

# Extreme Network OS Command Reference

Supporting Network OS 7.2.0c

## Legal Notice

Extreme Networks, Inc. reserves the right to make changes in specifications and other information contained in this document and its website without prior notice. The reader should in all cases consult representatives of Extreme Networks to determine whether any such changes have been made.

The hardware, firmware, software or any specifications described or referred to in this document are subject to change without notice.

## Trademarks

Extreme Networks and the Extreme Networks logo are trademarks or registered trademarks of Extreme Networks, Inc. in the United States and/or other countries.

All other names (including any product names) mentioned in this document are the property of their respective owners and may be trademarks or registered trademarks of their respective companies/owners.

For additional information on Extreme Networks trademarks, please see: [www.extremenetworks.com/company/legal/trademarks](http://www.extremenetworks.com/company/legal/trademarks)

## Open Source Declarations

Some software files have been licensed under certain open source or third-party licenses. End-user license agreements and open source declarations can be found at: [www.extremenetworks.com/support/policies/software-licensing](http://www.extremenetworks.com/support/policies/software-licensing)

# Contents

---

<b>Preface</b> .....	<b>39</b>
Document conventions.....	39
Notes, cautions, and warnings.....	39
Text formatting conventions.....	39
Command syntax conventions.....	40
Extreme resources.....	40
Document feedback.....	40
Contacting Extreme Technical Support.....	41
<b>About This Document</b> .....	<b>43</b>
Supported hardware and software.....	43
What's new in this document.....	43
New commands.....	43
Modified commands.....	43
Deprecated commands.....	44
New commands.....	44
Modified commands.....	44
Deprecated commands.....	45
<b>Using the Network OS CLI</b> .....	<b>47</b>
DCB command line interface.....	47
RBAC permissions .....	47
Default roles.....	47
Accessing the Network OS CLI through Telnet .....	48
Network OS CLI command modes.....	48
CLI keyboard shortcuts.....	54
Using the do command as a shortcut.....	55
Displaying CLI commands and command syntax.....	55
Completing CLI commands.....	56
Using CLI command output modifiers.....	56
Considerations for show command output .....	57
User-configurable VLAN IDs.....	57
Debug and system diagnostic commands.....	57
<b>Commands A through E</b> .....	<b>59</b>
aaa accounting.....	59
aaa authentication .....	61
accept-unicast-arp-request.....	63
access-group .....	64
access-list .....	66
action python-script.....	68
action-timeout.....	70
activate (NSX Controller connection profile).....	71
activate (OpenFlow).....	72
activate (OVSDB).....	74
activate (protected VLAG).....	75
activate (VXLAN gateway).....	76
address-family.....	77

address-family l2vpn evpn.....	78
address-family unicast (BGP).....	79
address-family unicast (VRF).....	81
advertise bgp-auto-nbr-tlv.....	82
advertise dcbx-fcoe-app-tlv .....	83
advertise dcbx-fcoe-logical-link-tlv .....	84
advertise dcbx-iscsi-app-tlv .....	85
advertise dcbx-tlv .....	86
advertise dot1-tlv .....	87
advertise dot3-tlv .....	88
advertise optional-tlv .....	89
advertise-backup .....	91
advertisement interval (fabric-map).....	92
advertisement-interval (VRRP).....	93
advertisement-interval-scale .....	94
ag .....	95
aggregate-address (BGP).....	96
alias .....	99
alias-config .....	100
allow non-profiled-macs .....	101
always-compare-med .....	102
always-propagate .....	103
area authentication (OSPFv3).....	104
area nssa (OSPFv2).....	106
area nssa (OSPFv3).....	109
area prefix-list (OSPFv2).....	111
area range (OSPFv2).....	113
area stub (OSPFv2).....	115
area virtual-link (OSPFv2).....	117
area virtual-link (OSPFv3).....	119
area virtual-link authentication (OSPFv3).....	121
arp .....	123
arp access-list.....	125
as-path-ignore .....	127
attach rbridge-id .....	128
attach rbridge-id (Fabric-Virtual-Gateway).....	130
attach vlan .....	131
auto-config-backup.....	133
auto-cost reference-bandwidth (OSPFv2).....	134
auto-cost reference-bandwidth (OSPFv3).....	136
auto-shutdown-new-neighbors.....	138
backup-advertisement-interval .....	140
banner incoming .....	141
banner login .....	142
banner motd .....	144
beacon.....	145
bfd.....	147
bfd holdover-interval.....	149
bfd interval.....	151
bfd shutdown.....	153



bfd-session-setup-delay.....	154
bfd-shutdown.....	155
bgp-redistribute-internal .....	156
bind .....	158
bp-rate-limit heavy module.....	160
bpdu-drop enable .....	162
bridge-priority .....	164
bsr-candidate.....	165
bsr-msg-interval.....	167
capability as4-enable .....	168
capture packet interface .....	169
cbs .....	172
cee .....	173
cee-map (configuration).....	174
certutil import ldapca .....	175
certutil import sshkey .....	177
certutil import syslogca .....	179
channel-group .....	181
chassis .....	183
chassis beacon .....	185
chassis disable .....	186
chassis enable .....	187
chassis fan airflow-direction .....	188
chassis power-cycle-db-shutdown.....	189
cidrecov .....	190
cipherset .....	193
cisco-interoperability .....	195
class .....	196
class-map .....	198
clear ag nport-utilization.....	199
clear arp .....	200
clear bfd counters .....	202
clear bgp evpn l2routes.....	204
clear bgp evpn local routes.....	206
clear bgp evpn mac-route dampening.....	207
clear bgp evpn neighbor.....	208
clear bgp evpn routes.....	210
clear counters access-list .....	212
clear counters all.....	214
clear counters interface .....	215
clear counters slot-id .....	217
clear counters storm-control .....	218
clear dot1x statistics .....	220
clear dot1x statistics interface .....	221
clear edge-loop-detection .....	222
clear ip arp inspection statistics.....	223
clear ip arp suppression-cache.....	224
clear ip arp suppression-statistics.....	225
clear ip bgp dampening .....	226
clear ip bgp flap-statistics .....	227

clear ip bgp local routes .....	229
clear ip bgp neighbor .....	230
clear ip bgp routes .....	232
clear ip bgp traffic .....	233
clear ip dhcp relay statistics .....	234
clear ip fabric-virtual-gateway.....	236
clear ip igmp groups .....	237
clear ip igmp statistics interface .....	239
clear ip ospf .....	241
clear ip pim mcache .....	243
clear ip pim rp-map .....	244
clear ip pim traffic .....	245
clear ip route .....	246
clear ipv6 bgp dampening .....	248
clear ipv6 bgp flap-statistics .....	249
clear ipv6 bgp local routes .....	251
clear ipv6 bgp neighbor.....	252
clear ipv6 bgp routes.....	254
clear ipv6 bgp traffic.....	255
clear ipv6 counters .....	256
clear ipv6 dhcp relay statistics .....	257
clear ipv6 fabric-virtual-gateway.....	258
clear ipv6 mld groups .....	259
clear ipv6 mld statistics .....	260
clear ipv6 nd suppression-cache.....	262
clear ipv6 nd suppression-statistics.....	263
clear ipv6 neighbor .....	264
clear ipv6 ospf .....	266
clear ipv6 route .....	269
clear ipv6 vrrp statistics .....	271
clear lacp .....	273
clear lacp counters .....	274
clear lldp neighbors .....	275
clear lldp statistics .....	276
clear logging auditlog .....	277
clear logging raslog .....	278
clear mac-address-table conversational.....	280
clear mac-address-table dynamic .....	281
clear maps dashboard.....	282
clear nas statistics .....	283
clear openflow .....	284
clear overlay-gateway .....	285
clear policy-map-counters .....	286
clear sessions .....	287
clear sflow statistics .....	288
clear spanning-tree counter .....	289
clear spanning-tree detected-protocols .....	291
clear statistics openflow .....	293
clear support .....	294
clear udd statistics .....	295

clear vrrp statistics .....	296
CLI.....	298
client-to-client-reflection .....	301
clock set .....	303
clock timezone (Privileged EXEC mode) .....	304
clock timezone (RBridge ID configuration mode).....	306
cluster-id .....	308
compare-med-empty-aspath .....	309
compare-routerid .....	310
confederation identifier.....	311
confederation peers.....	312
configure terminal .....	313
conform-set-dscp .....	314
conform-set-prec .....	315
conform-set-tc .....	316
connector-group.....	317
connector .....	319
continue .....	320
controller.....	321
copy .....	322
copy snapshot .....	325
copy support .....	327
copy support-interactive .....	329
cos-mutation .....	331
counter reliability .....	332
crypto ca authenticate.....	333
crypto ca enroll.....	335
crypto ca import.....	337
crypto ca trustpoint.....	339
crypto key .....	340
custom-profile.....	342
dampening .....	344
database-overflow-interval (OSPFv2).....	346
database-overflow-interval (OSPFv3).....	348
debug access-list-log buffer .....	350
debug bfd .....	351
debug dhcp packet buffer .....	353
debug dhcp packet buffer clear .....	355
debug dhcp packet buffer interface .....	356
debug ip .....	358
debug ip bgp .....	360
debug ip bgp neighbor .....	362
debug ip fabric-virtual-gateway.....	363
debug ip igmp .....	365
debug ip ospf .....	367
debug ip pim .....	369
debug ip rtm .....	371
debug ip vrf .....	373
debug ipv6 bgp.....	374
debug ipv6 bgp neighbor.....	376

debug ipv6 dhcpv6 packet buffer.....	378
debug ipv6 mld.....	380
debug ipv6 nd.....	382
debug ipv6 ospf .....	383
debug ipv6 packet buffer.....	386
debug lacp .....	388
debug lldp dump .....	390
debug lldp packet .....	392
debug show bp-stats interface.....	394
debug show qos drop-reason interface.....	396
debug spanning-tree .....	398
debug udld packet .....	400
debug vrrp .....	402
default-behavior .....	404
default-config enable.....	405
default-information-originate (BGP).....	406
default-information-originate (OSPFv2).....	407
default-information-originate (OSPFv3).....	409
default-local-preference .....	411
default-metric (BGP).....	412
default-metric (OSPF).....	414
default-passive-interface .....	415
delay.....	416
delete .....	417
description (event-handler).....	418
description (interfaces).....	419
description (LLDP).....	420
description (Port Mirroring).....	421
description (VRRP).....	422
destination .....	423
device-connectivity.....	424
dhcp auto-deployment enable .....	425
dhcpd enable.....	427
dhcpd restart.....	428
diag burninerrclear .....	429
diag clearerror .....	430
diag dport-test interface.....	431
diag portledtest .....	433
diag portloopbacktest .....	436
diag post enable .....	438
diag prbstest .....	439
diag setcycle .....	441
diag systemverification .....	443
diag turboramtest .....	445
dir .....	447
disable (Fabric-Virtual-Gateway).....	448
distance (BGP).....	449
distance (OSPF).....	450
distribute-list route-map .....	452
distribute-list prefix-list (OSPFv3).....	453

dot1x authentication .....	454
dot1x enable .....	455
dot1x mac-auth-bypass.....	456
dot1x mac-auth-enable.....	458
dot1x port-control .....	459
dot1x quiet-period .....	460
dot1x reauthenticate .....	461
dot1x reauthentication .....	462
dot1x reauthMax .....	463
dot1x test eapol-capable .....	464
dot1x test timeout .....	466
dot1x timeout re-authperiod .....	467
dot1x timeout server-timeout .....	468
dot1x timeout supp-timeout .....	469
dot1x timeout tx-period .....	470
dpod .....	471
dscp-cos .....	473
dscp-mutation .....	474
dscp-traffic-class .....	475
duplicate-mac-timer.....	476
ebs .....	478
edge-loop-detection port-priority .....	479
edge-loop-detection vlan .....	480
eir .....	481
email.....	482
enable (Access Gateway) .....	483
enable (Fabric-Virtual-Gateway).....	484
enable (MAPS) .....	486
enable (VRRP-E) .....	488
enable statistics direction .....	489
end .....	491
enforce-first-as .....	492
error-disable-timeout enable .....	493
error-disable-timeout interval .....	494
esi .....	496
event-handler.....	498
event-handler abort action.....	500
event-handler activate.....	501
evpn-instance.....	504
exceed-set-dscp .....	506
exceed-set-prec .....	507
exceed-set-tc .....	508
execute-script.....	509
exit .....	515
export map evpn.....	516
extend vlan .....	518
external-lsdb-limit (OSPFv2).....	519
external-lsdb-limit (OSPFv3).....	520
<b>Commands F through O.....</b>	<b>521</b>
fabric dport mode.....	521

fabric ecmp load-balance .....	523
fabric ecmp load-balance-hash-swap .....	525
fabric isl enable .....	526
fabric login-policy.....	527
fabric neighbor-discovery disable .....	528
fabric port-channel.....	529
fabric route mcast .....	531
fabric trunk enable .....	532
fabric-map .....	533
fast-external-fallover .....	535
fastboot .....	536
fcf-group.....	537
fcmap .....	539
fcoe .....	540
fcoe binding.....	541
fcoe-enodes .....	542
fcoe-profile (AMPP).....	543
fcoeport .....	544
fcport-group.....	545
fcport-group-rbid.....	546
fcsp auth .....	547
fcsp auth-secret dhchap .....	549
filter-change-update-delay .....	551
fips selftests .....	552
fips zeroize .....	554
firmware activate .....	556
firmware commit .....	557
firmware download .....	558
firmware download ftp .....	561
firmware download interactive .....	563
firmware download logical-chassis .....	565
firmware download scp .....	568
firmware download sftp .....	570
firmware download tftp .....	572
firmware download usb .....	574
firmware install .....	576
firmware peripheral-update microcode .....	578
firmware recover .....	580
firmware restore .....	581
firmware sync .....	582
flexport.....	583
forward-delay .....	584
gateway-address.....	586
gateway-mac-address.....	587
graceful-restart (BGP).....	588
graceful-restart (OSPFv2).....	591
graceful-restart helper (OSPFv3).....	593
graceful-shutdown .....	594
gratuitous-arp timer.....	596
group.....	598

ha chassisreboot .....	600
ha disable .....	601
ha enable .....	602
ha failover .....	604
ha sync start .....	605
ha sync stop .....	606
hardware .....	607
hardware-profile.....	608
hardware-profile vlan-classification.....	611
hello (LLDP).....	613
hello (UDLD).....	614
hello-interval .....	615
hello-interval (ELD).....	617
hello-time .....	618
hello-timer .....	620
hold-time .....	621
hold-time (Fabric-Virtual-Gateway).....	622
host-table aging-mode conversational.....	623
host-table aging-time conversational.....	624
http server .....	625
import map evpn.....	628
inactivity-timer .....	630
install-igp-cost .....	631
instance .....	632
interface .....	634
interface (range specification).....	636
interface loopback .....	639
interface management .....	640
interface ve .....	642
interface vlan .....	643
interval.....	645
ip access-group .....	647
ip access-list .....	649
ip address .....	651
ip address (NSX controller configuration).....	653
ip address (OpenFlow).....	654
ip address (VXLAN).....	656
ip anycast-address.....	657
ip anycast-gateway-mac.....	658
ip arp-aging-timeout .....	660
ip arp inspection.....	661
ip arp inspection filter.....	662
ip arp inspection logging acl-match.....	663
ip arp inspection trust.....	665
ip arp learn-any.....	667
ip as-path access-list .....	668
ip community-list extended .....	669
ip community-list standard .....	671
ip dhcp relay address .....	673
ip dhcp relay gateway address.....	675

ip dhcp relay information option.....	677
ip directed-broadcast .....	679
ip dns .....	680
ip extcommunity-list.....	681
ip fabric-virtual-gateway.....	683
ip icmp address-mask.....	684
ip icmp echo-reply .....	685
ip icmp rate-limiting .....	686
ip icmp redirect.....	687
ip icmp unreachable .....	688
ip igmp immediate-leave .....	689
ip igmp last-member-query-count.....	690
ip igmp last-member-query-interval .....	691
ip igmp query-interval .....	692
ip igmp query-max-response-time .....	693
ip igmp snooping enable (global version).....	694
ip igmp snooping enable .....	695
ip igmp snooping fast-leave .....	696
ip igmp snooping mrouter interface .....	697
ip igmp snooping querier enable .....	698
ip igmp snooping restrict-unknown-multicast .....	699
ip igmp snooping robustness-variable.....	700
ip igmp startup-query-count.....	701
ip igmp startup-query-interval.....	702
ip igmp static-group .....	703
ip import routes (IPv4 VRF address-family configuration mode).....	705
ip import routes (RBridge ID configuration mode).....	706
ip interface .....	707
ip mtu .....	709
ip multicast-boundary .....	711
ip ospf active .....	712
ip ospf area .....	713
ip ospf auth-change-wait-time .....	714
ip ospf authentication-key .....	716
ip ospf bfd .....	718
ip ospf cost .....	719
ip ospf database-filter .....	720
ip ospf dead-interval .....	722
ip ospf hello-interval .....	723
ip ospf md5-authentication .....	725
ip ospf mtu-ignore .....	727
ip ospf network .....	728
ip ospf passive .....	729
ip ospf priority .....	730
ip ospf retransmit-interval .....	731
ip ospf transmit-delay .....	732
ip pim dr-priority .....	733
ip pim multinet enable.....	734
ip pim neighbor-filter.....	735
ip pim-sparse .....	737



ip policy route-map .....	738
ip prefix-list .....	739
ip proxy-arp .....	741
ip receive access-group.....	742
ip route .....	744
ip route next-hop-vrf .....	746
ip route static bfd .....	748
ip route static bfd holdover-interval .....	750
ip router-id .....	752
ip unnumbered.....	753
ipv6 access-group .....	755
ipv6 access-list .....	757
ipv6 address .....	759
ipv6 address anycast .....	760
ipv6 address eui-64 .....	761
ipv6 address link-local .....	762
ipv6 address use-link-local-only .....	763
ipv6 anycast-address.....	764
ipv6 anycast-gateway-mac.....	765
ipv6 dhcp relay address .....	767
ipv6 fabric-virtual-gateway.....	769
ipv6 icmpv6 echo-reply .....	770
ipv6 icmpv6 rate-limiting .....	771
ipv6 icmpv6 redirect.....	772
ipv6 icmpv6 unreachable.....	773
ipv6 import routes (IPv6 VRF address-family configuration mode).....	774
ipv6 import routes (RBridge ID configuration mode).....	775
ipv6 mld last-member-query-count .....	776
ipv6 mld last-member-query-interval .....	777
ipv6 mld query-interval .....	778
ipv6 mld query-max-response-time .....	779
ipv6 mld snooping enable .....	780
ipv6 mld snooping fast-leave .....	781
ipv6 mld snooping mrouter interface .....	782
ipv6 mld snooping querier enable .....	783
ipv6 mld snooping restrict-unknown-multicast .....	784
ipv6 mld snooping robustness-variable .....	785
ipv6 mld startup-query-count .....	786
ipv6 mld startup-query-interval .....	787
ipv6 mld static-group interface .....	788
ipv6 mtu .....	789
ipv6 nd cache expire .....	790
ipv6 nd dad attempts .....	791
ipv6 nd dad time .....	792
ipv6 nd hoplimit .....	793
ipv6 nd managed-config-flag .....	794
ipv6 nd mtu .....	795
ipv6 nd ns-interval .....	796
ipv6 nd other-config-flag .....	797
ipv6 nd prefix .....	798

ipv6 nd ra-interval .....	799
ipv6 nd ra-lifetime .....	800
ipv6 nd reachable-time .....	801
ipv6 nd retrans-timer .....	802
ipv6 nd suppress-ra .....	803
ipv6 neighbor .....	804
ipv6 ospf active .....	805
ipv6 ospf area .....	806
ipv6 ospf authentication ipsec .....	808
ipv6 ospf authentication ipsec disable .....	809
ipv6 ospf authentication spi.....	810
ipv6 ospf bfd .....	812
ipv6 ospf cost .....	813
ipv6 ospf dead-interval .....	815
ipv6 ospf hello-interval .....	817
ipv6 ospf hello-jitter .....	819
ipv6 ospf instance .....	820
ipv6 ospf mtu-ignore .....	821
ipv6 ospf network .....	822
ipv6 ospf passive .....	824
ipv6 ospf priority .....	825
ipv6 ospf retransmit-interval .....	827
ipv6 ospf suppress-linklsa .....	828
ipv6 ospf transmit-delay .....	829
ipv6 prefix-list.....	830
ipv6 protocol vrrp .....	832
ipv6 protocol vrrp-extended .....	833
ipv6 rguard .....	834
ipv6 receive access-group.....	835
ipv6 route .....	837
ipv6 route static bfd .....	840
ipv6 route static bfd holdover-interval .....	842
ipv6 router ospf .....	844
ipv6 unreachable .....	845
ipv6 vrrp-extended-group .....	846
ipv6 vrrp-group .....	847
ipv6 vrrp-suppress-interface-ra .....	848
iscsi-priority .....	849
isl_r_rdy .....	850
iterations.....	851
keep-alive timeout (fabric-map).....	852
key-add-remove-interval.....	853
key-rollover-interval.....	854
keypair.....	856
l2traceroute .....	857
lACP default-up .....	859
lACP port-priority .....	860
lACP system-priority .....	861
lACP timeout .....	862
ldap-server host .....	863

ldap-server maprole .....	865
license add .....	866
license remove .....	868
listen-limit.....	870
listen-range .....	872
line vty exec-timeout .....	874
linecard .....	876
lldp dcbx-version .....	878
lldp disable .....	879
lldp iscsi-priority .....	880
lldp profile .....	881
load-balance .....	882
load-balancing.....	884
load-balancing-disable.....	885
local-as .....	886
log (OSPFv2).....	887
log (OSPFv3).....	889
log-dampening-debug .....	891
logging auditlog class .....	892
logging raslog console .....	893
logging syslog-client.....	895
logging syslog-facility local .....	896
logging syslog-server .....	897
logical-chassis principal-priority .....	899
logical-chassis principal-switchover .....	900
long-distance-isl .....	901
mac .....	904
mac access-group .....	905
mac access-list standard .....	907
mac-address-reduction .....	908
mac-address-table .....	909
mac-address-table consistency-check interval.....	912
mac-address-table consistency-check suppress .....	914
mac-address-table mac-move action.....	915
mac-address-table mac-move auto-recovery.....	917
mac-address-table mac-move detect .....	919
mac-address-table mac-move limit .....	920
mac-group .....	922
mac-learning disable vlan.....	923
mac-learning protocol bgp.....	925
mac-rebalance .....	926
mac-refresh.....	927
management .....	929
map fport interface fcoe .....	930
map qos .....	931
map sflow .....	932
map vlan .....	933
maps.....	935
maps reapply-policy.....	936
match .....	937

match (route map).....	938
match access-list .....	942
match as-path .....	943
match community .....	944
match extcommunity.....	945
match interface .....	946
match ip address .....	948
match ip next-hop .....	949
match ipv6 address .....	950
match metric .....	951
match protocol bgp .....	952
match route-type .....	953
match tag .....	954
max-age .....	955
max-hops .....	957
max-mcache .....	958
max-metric router-lsa .....	959
max-metric router-lsa (OSPFv3).....	961
max-route .....	964
maxas-limit .....	965
maximum-paths (BGP).....	967
maximum-paths (OSPF).....	969
maximum-paths ebgp ibgp .....	970
med-missing-as-worst .....	972
message-interval .....	973
metric-type .....	974
minimum-links .....	976
mode (LLDP) .....	977
mode (27x40 GbE line card).....	978
mode (VDX6940-144S).....	979
modes .....	981
monitor session .....	983
monitor session (VXLAN).....	984
mtrace.....	986
mtu .....	988
multipath .....	989
multiplier (LLDP).....	991
multiplier (UDLD).....	992
name (VLAN interfaces).....	993
nas auto-qos.....	994
nas server-ip.....	995
nbr-timeout .....	996
neighbor (OSPF).....	997
neighbor accept-ldp-neighbors .....	998
neighbor activate.....	999
neighbor additional-paths advertise .....	1001
neighbor advertisement-interval .....	1003
neighbor allowas-in .....	1005
neighbor alternate-as .....	1007
neighbor as-override .....	1009

neighbor bfd .....	1011
neighbor capability additional-paths .....	1013
neighbor capability as4 .....	1015
neighbor capability orf prefixlist.....	1017
neighbor default-originate .....	1019
neighbor description .....	1021
neighbor ebgp-btsh .....	1023
neighbor ebgp-multihop .....	1025
neighbor enable-peer-as-check.....	1027
neighbor enforce-first-as .....	1029
neighbor filter-list .....	1031
neighbor graceful-shutdown .....	1033
neighbor local-as .....	1035
neighbor maxas-limit in .....	1037
neighbor maximum-prefix .....	1039
neighbor next-hop-self .....	1041
neighbor next-hop-unchanged.....	1043
neighbor password .....	1045
neighbor peer-group .....	1047
neighbor prefix-list .....	1049
neighbor remote-as .....	1051
neighbor remove-private-as.....	1053
neighbor route-map .....	1055
neighbor route-reflector-client .....	1057
neighbor send-community .....	1059
neighbor shutdown .....	1061
neighbor soft-reconfiguration inbound .....	1063
neighbor static-network-edge.....	1065
neighbor timers .....	1066
neighbor unsuppress-map .....	1068
neighbor update-source .....	1070
neighbor weight .....	1072
network .....	1074
next-hop-enable-default .....	1076
next-hop-recursion .....	1077
nonstop-routing (OSPF).....	1078
nport interface fiberchannel (ag).....	1079
nport interface fibrechannel (port group).....	1080
nsx-controller client-cert .....	1081
nsx-controller name .....	1083
ntp authentication-key .....	1084
ntp server .....	1086
ntp source-ip.....	1088
openflow-controller.....	1090
openflow enable .....	1092
openflow logical-instance.....	1094
oscmd .....	1096
overlay-gateway .....	1099
ovsdb-server.....	1102
<b>Commands P through short-path-forwarding.....</b>	<b>1103</b>

passive.....	1103
password-attributes .....	1105
pdu-rx-limit .....	1108
permit ip host.....	1109
pg .....	1111
ping .....	1113
police cir .....	1116
police-priority-map .....	1117
policy.....	1119
policy-map .....	1120
port (OVSDB).....	1121
port-channel .....	1122
port-channel-redundancy-group .....	1123
port-channel path-cost .....	1124
port-group .....	1126
port-profile (global configuration mode).....	1127
port-profile (port-profile-domain configuration mode).....	1128
port-profile-domain .....	1130
port-profile-port .....	1131
power-off .....	1133
power-off linecard .....	1134
power-on .....	1135
power-on linecard .....	1136
precedence .....	1137
preempt-mode .....	1138
preprovision.....	1139
priority .....	1141
priority-group-table .....	1143
priority-tag .....	1145
private-vlan .....	1146
private-vlan association .....	1147
profile (LLDP) .....	1148
prom-access disable .....	1149
protect-mode enable .....	1150
protected-port enable.....	1151
protocol edge-loop-detection .....	1152
protocol lldp .....	1153
protocol spanning-tree .....	1154
protocol udld .....	1156
protocol vrrp .....	1157
protocol vrrp-extended .....	1158
pwd .....	1159
python.....	1160
qos.....	1164
qos cos .....	1166
qos cos-mutation .....	1168
qos drop-monitor enable.....	1170
qos dscp-cos .....	1171
qos dscp-mutation .....	1172
qos dscp-traffic-class .....	1174

qos flowcontrol .....	1176
qos map cos-mutation .....	1178
qos map dscp-cos .....	1180
qos map dscp-mutation .....	1182
qos map dscp-traffic-class .....	1184
qos random-detect traffic-class .....	1186
qos rcv-queue limit.....	1187
qos red profile .....	1188
qos service-policy.....	1190
qos tx-queue limit.....	1191
qos-profile (AMPP).....	1192
radius-server .....	1193
rasman .....	1195
rate-limit-delay get netconf .....	1197
rate-limit-delay set netconf .....	1198
rbridge-id .....	1199
rd (VNI).....	1200
rd (VRF).....	1202
rd auto (EVPN).....	1204
reconnect-interval .....	1205
redistribute .....	1206
region .....	1209
relay.....	1210
reload .....	1211
remap fabric-priority .....	1213
remap lossless-priority .....	1214
rename .....	1215
rename (Access Gateway mode).....	1216
resequence access-list .....	1217
reserved-vlan .....	1219
resource-monitor cpu enable .....	1220
resource-monitor memory .....	1221
resource-monitor process memory .....	1223
restrict-flooding .....	1225
retain route-target all .....	1226
revision .....	1227
rfc1583-compatibility (OSPF).....	1228
rfc1587-compatibility (OSPF).....	1229
rib-route-limit .....	1230
rmon alarm .....	1232
rmon collection history .....	1234
rmon collection stats .....	1236
rmon event .....	1237
role name .....	1238
root access console.....	1240
root enable.....	1241
route-map .....	1242
router bgp .....	1244
router fabric-virtual-gateway.....	1245
router ospf .....	1246

router pim .....	1247
route-target (EVPN).....	1248
route-target (VNI).....	1250
route-target (VRF).....	1252
rp-address .....	1254
rp-adv-interval.....	1256
rp-candidate.....	1257
rpf-mode.....	1259
rspan-vlan .....	1261
rule .....	1262
rule (MAPS).....	1264
run-mode.....	1267
scheduler .....	1269
script reload.....	1270
scp.....	1271
secpolicy activate .....	1273
secpolicy defined-policy .....	1275
security-profile (AMPP).....	1277
seq (IPv4 extended ACLs).....	1278
seq (IPv6 extended ACLs).....	1283
seq (IPv4 standard ACLs).....	1288
seq (IPv6 standard ACLs).....	1290
seq (MAC extended ACLs).....	1292
seq (MAC standard ACLs).....	1295
service password-encryption .....	1297
service-policy .....	1298
set as-path .....	1300
set as-path prepend .....	1301
set automatic-tag .....	1304
set comm-list .....	1305
set community .....	1306
set cos traffic-class .....	1307
set dampening .....	1308
set distance .....	1309
set dscp .....	1310
set extcommunity.....	1311
set ip interface null0 .....	1313
set ip next-hop .....	1314
set ipv6 next-hop .....	1315
set local-preference .....	1316
set metric .....	1317
set metric-type .....	1318
set origin .....	1319
set-priority .....	1320
set route-type .....	1321
set tag .....	1322
set weight .....	1323
sflow (VXLAN).....	1324
sflow collector .....	1326
sflow enable (global version).....	1328



sflow enable (interface version).....	1329
sflow polling-interval (global version).....	1330
sflow polling-interval (interface version).....	1331
sflow sample-rate (global version).....	1332
sflow sample-rate (interface version).....	1333
sflow source-ip.....	1334
sflow-profile .....	1336
sflow profile-map.....	1337
sfp breakout .....	1338
shape .....	1339
short-path-forwarding .....	1340
<b>Show commands.....</b>	<b>1341</b>
show access-list.....	1341
show access-list-log buffer .....	1345
show access-list-log buffer config.....	1347
show ag .....	1348
show ag map .....	1350
show ag nport-utilization.....	1352
show ag pg .....	1354
show arp .....	1356
show arp access-list.....	1359
show bare-metal.....	1360
show bfd.....	1361
show bfd neighbors.....	1364
show bfd neighbors application.....	1366
show bfd neighbors dest-ip.....	1369
show bfd neighbors details.....	1371
show bfd neighbors interface.....	1374
show bfd neighbors vrf.....	1377
show bgp evpn dampened-routes.....	1379
show bgp evpn interface port-channel.....	1380
show bgp evpn interface tunnel.....	1381
show bgp evpn l2route detail.....	1382
show bgp evpn l2route next-hop.....	1385
show bgp evpn l2route summary.....	1389
show bgp evpn l2route type arp.....	1392
show bgp evpn l2route type auto-discovery.....	1396
show bgp evpn l2route type ethernet-segment.....	1398
show bgp evpn l2route type inclusive-multicast.....	1400
show bgp evpn l2route type mac.....	1404
show bgp evpn l2route type nd.....	1408
show bgp evpn l2route unreachable.....	1411
show bgp evpn l3vni.....	1413
show bgp evpn neighbors .....	1416
show bgp evpn neighbors advertised-routes detail.....	1419
show bgp evpn neighbors advertised-routes type.....	1422
show bgp evpn neighbors routes best.....	1425
show bgp evpn neighbors routes detail.....	1428
show bgp evpn neighbors routes not-installed-best.....	1431
show bgp evpn neighbors routes type.....	1432

show bgp evpn neighbors routes unreachable.....	1435
show bgp evpn neighbors routes-summary.....	1437
show bgp evpn routes.....	1438
show bgp evpn routes best.....	1439
show bgp evpn routes detail.....	1442
show bgp evpn routes local.....	1444
show bgp evpn routes next-hop.....	1447
show bgp evpn routes no-best.....	1450
show bgp evpn routes not-installed-best.....	1452
show bgp evpn routes rd.....	1453
show bgp evpn routes rd type.....	1455
show bgp evpn routes summary.....	1460
show bgp evpn routes type arp.....	1463
show bgp evpn routes type auto-discovery.....	1466
show bgp evpn routes type ethernet-segment.....	1468
show bgp evpn routes type inclusive-multicast.....	1470
show bgp evpn routes type ipv4-prefix.....	1475
show bgp evpn routes type ipv6-prefix.....	1479
show bgp evpn routes type mac.....	1482
show bgp evpn routes type nd.....	1485
show bgp evpn routes unreachable.....	1488
show bgp evpn summary.....	1490
show bpdu-drop .....	1491
show capture packet interface .....	1492
show cee maps .....	1495
show cert-util ldapca .....	1496
show cert-util sshkey .....	1497
show cert-util syslogca .....	1498
show chassis .....	1499
show cipherset .....	1503
show class-maps .....	1504
show cli .....	1505
show cli history .....	1506
show clock .....	1507
show config snapshot .....	1508
show copy-support status .....	1509
show crypto ca .....	1510
show crypto key.....	1512
show dadstatus .....	1513
show debug dhcp packet .....	1514
show debug dhcp packet buffer .....	1515
show debug ip bgp all .....	1518
show debug ip igmp .....	1519
show debug ip pim .....	1520
show debug ipv6 mld .....	1521
show debug ipv6 packet.....	1523
show debug lacp .....	1525
show debug lldp .....	1526
show debug spanning-tree .....	1527
show debug uddl .....	1528

show debug vrrp .....	1529
show default threshold .....	1530
show default-vlan .....	1532
show dhcpd configuration.....	1533
show dpod .....	1534
show diag burninerrshow .....	1536
show diag burninerrshowerrLog .....	1537
show diag burninstatus .....	1539
show diag post results .....	1540
show diag setcycle .....	1542
show diag status .....	1543
show dot1x .....	1544
show dot1x all .....	1545
show dot1x diagnostics interface .....	1546
show dot1x interface .....	1547
show dot1x session-info interface .....	1549
show dot1x statistics interface .....	1550
show dpod .....	1551
show dport-test.....	1553
show edge-loop-detection detail .....	1556
show edge-loop-detection globals .....	1557
show edge-loop-detection interface .....	1558
show edge-loop-detection rbridge-id .....	1560
show environment fan .....	1561
show environment history .....	1562
show environment power .....	1565
show environment sensor .....	1566
show environment temp .....	1567
show event-handler activations.....	1569
show fabric all .....	1571
show fabric ecmp group.....	1573
show fabric ecmp load-balance .....	1575
show fabric isl .....	1576
show fabric islports .....	1579
show fabric login-policy.....	1583
show fabric port-channel .....	1585
show fabric route linkinfo .....	1586
show fabric route multicast .....	1590
show fabric route neighbor-state .....	1593
show fabric route pathinfo .....	1596
show fabric route topology .....	1604
show fabric trunk .....	1607
show fcoe devices.....	1610
show fcoe fabric-map .....	1611
show fcoe fcf-group.....	1613
show fcoe fcoe-enodes.....	1614
show fcoe fcoe-map.....	1615
show fcoe fcport-group.....	1616
show fcoe interface .....	1617
show fcoe login .....	1618

show fcsp auth-secret dh-chap .....	1620
show fibrechannel login .....	1621
show file .....	1623
show fips .....	1625
show firmwaredownloadhistory .....	1626
show firmwaredownloadstatus .....	1627
show global-running-config .....	1629
show ha .....	1632
show hardware port-group.....	1634
show hardware connector-group.....	1635
show hardware-profile.....	1636
show history .....	1640
show http server status.....	1641
show interface .....	1642
show interface description .....	1646
show interface FibreChannel .....	1647
show interface management .....	1652
show interface stats .....	1654
show interface status .....	1657
show interface trunk .....	1658
show inventory .....	1659
show ip anycast-gateway.....	1660
show ip arp inspection.....	1662
show ip arp inspection interfaces.....	1664
show ip arp inspection statistics.....	1666
show ip arp suppression-cache.....	1668
show ip arp suppression-statistics.....	1669
show ip arp suppression-status.....	1670
show ip as-path-list.....	1671
show ip bgp.....	1672
show ip bgp attribute-entries .....	1674
show ip bgp dampened-paths .....	1676
show ip bgp filtered-routes .....	1677
show ip bgp flap-statistics .....	1679
show ip bgp neighbors .....	1681
show ip bgp neighbors advertised-routes .....	1687
show ip bgp neighbors flap-statistics .....	1689
show ip bgp neighbors last-packet-with-error.....	1690
show ip bgp neighbors received .....	1691
show ip bgp neighbors received-routes .....	1692
show ip bgp neighbors routes .....	1694
show ip bgp neighbors routes-summary .....	1696
show ip bgp peer-group .....	1698
show ip bgp rbridge-id.....	1701
show ip bgp routes .....	1702
show ip bgp routes community .....	1708
show ip bgp summary .....	1710
show ip community-list.....	1712
show ip dhcp relay address interface .....	1713
show ip dhcp relay address rbridge-id .....	1716

show ip dhcp relay gateway.....	1718
show ip dhcp relay statistics .....	1720
show ip dns.....	1722
show ip extcommunity-list.....	1723
show ip fabric-virtual-gateway.....	1724
show ip igmp groups .....	1726
show ip igmp interface .....	1728
show ip igmp snooping .....	1730
show ip igmp static-groups.....	1731
show ip igmp statistics interface .....	1734
show ip interface .....	1736
show ip interface loopback .....	1740
show ip interface ve .....	1741
show ip next-hop.....	1742
show ip ospf .....	1743
show ip ospf area .....	1746
show ip ospf border-routers .....	1749
show ip ospf config .....	1750
show ip ospf database .....	1752
show ip ospf filtered-lsa area .....	1755
show ip ospf interface .....	1757
show ip ospf neighbor .....	1759
show ip ospf redistribute route .....	1761
show ip ospf routes .....	1762
show ip ospf summary .....	1763
show ip ospf traffic .....	1764
show ip ospf virtual .....	1766
show ip pim bsr .....	1768
show ip pim group .....	1770
show ip pim mcache .....	1772
show ip pim neighbor .....	1774
show ip pim rpf .....	1777
show ip pim rp-hash .....	1779
show ip pim rp-map .....	1781
show ip pim rp-set .....	1783
show ip pim-sparse .....	1785
show ip pim traffic .....	1788
show ip prefix-list.....	1790
show ip route .....	1791
show ip route import.....	1796
show ip route system-summary .....	1798
show ip route-map.....	1800
show ip static route.....	1801
show ipv6 anycast-gateway.....	1802
show ipv6 bgp attribute-entries.....	1804
show ipv6 bgp dampened-paths.....	1806
show ipv6 bgp filtered-routes.....	1807
show ipv6 bgp filtered-routes detail.....	1810
show ipv6 bgp flap-statistics.....	1813
show ipv6 bgp neighbors.....	1815

show ipv6 bgp neighbors advertised-routes.....	1817
show ipv6 bgp neighbors flap-statistics.....	1821
show ipv6 bgp neighbors last-packet-with-error.....	1823
show ipv6 bgp neighbors rbridge-id.....	1825
show ipv6 bgp neighbors received.....	1826
show ipv6 bgp neighbors received-routes.....	1828
show ipv6 bgp neighbors rib-out-routes.....	1831
show ipv6 bgp neighbors routes.....	1833
show ipv6 bgp neighbors routes-summary.....	1835
show ipv6 bgp peer-group.....	1838
show ipv6 bgp rbridge-id.....	1840
show ipv6 bgp routes.....	1841
show ipv6 bgp routes community .....	1845
show ipv6 bgp summary.....	1847
show ipv6 counters interface .....	1851
show ipv6 dhcp relay address interface .....	1853
show ipv6 dhcp relay address rbridge-id .....	1855
show ipv6 dhcp relay statistics .....	1857
show ipv6 fabric-virtual-gateway.....	1859
show ipv6 interface .....	1861
show ipv6 mld groups .....	1864
show ipv6 mld interface.....	1866
show ipv6 mld snooping .....	1867
show ipv6 mld statistics .....	1868
show ipv6 nd interface .....	1869
show ipv6 nd suppression-cache.....	1872
show ipv6 nd suppression-statistics.....	1873
show ipv6 nd suppression-status.....	1874
show ipv6 neighbor .....	1875
show ipv6 ospf .....	1877
show ipv6 ospf area .....	1879
show ipv6 ospf database .....	1880
show ipv6 ospf interface .....	1886
show ipv6 ospf memory .....	1889
show ipv6 ospf neighbor .....	1891
show ipv6 ospf redistribute route .....	1894
show ipv6 ospf routes .....	1895
show ipv6 ospf spf .....	1897
show ipv6 ospf summary .....	1899
show ipv6 ospf virtual-links .....	1900
show ipv6 ospf virtual-neighbor .....	1902
show ipv6 prefix-list .....	1904
show ipv6 rguard.....	1905
show ipv6 route .....	1907
show ipv6 static route .....	1911
show ipv6 vrrp .....	1912
show lacp .....	1919
show lacp sys-id .....	1920
show license .....	1921
show license id .....	1923

show linecard .....	1924
show lldp .....	1926
show lldp interface .....	1927
show lldp neighbors .....	1929
show lldp statistics .....	1932
show logging auditlog .....	1933
show logging raslog .....	1934
show mac-address-table .....	1936
show mac-address-table consistency-check.....	1940
show mac-address-table count evpn.....	1941
show mac-address-table evpn.....	1942
show mac-address-table mac-move.....	1943
show maps dashboard.....	1944
show maps group.....	1947
show maps policy.....	1949
show media .....	1951
show media interface .....	1952
show media linecard .....	1955
show media optical-monitoring interface.....	1958
show media tunable-optic-sfpp.....	1960
show mgmt-ip-service.....	1962
show mm .....	1963
show monitor .....	1965
show name-server brief .....	1966
show name-server detail .....	1967
show name-server nodefind .....	1969
show name-server zonemember .....	1971
show nas statistics.....	1973
show netconf client-capabilities .....	1974
show netconf-state capabilities .....	1975
show netconf-state datastores .....	1976
show netconf-state schemas .....	1977
show netconf-state sessions .....	1978
show netconf-state statistics .....	1979
show notification stream .....	1980
show nsx-controller .....	1981
show ntp status .....	1983
show openflow .....	1984
show openflow controller.....	1987
show openflow flow .....	1988
show openflow group .....	1991
show openflow interface .....	1993
show openflow meter .....	1995
show openflow queues .....	1997
show openflow resources .....	1999
show overlapping-vlan-resource usage .....	2000
show overlay-gateway .....	2001
show ovsdb-server.....	2003
show policymap .....	2004
show port port-channel .....	2006

show port-channel .....	2007
show port-channel-redundancy-group .....	2009
show port-profile .....	2010
show port-profile domain .....	2011
show port-profile interface .....	2013
show port-profile name .....	2014
show port-security .....	2015
show port-security addresses .....	2016
show port-security interface .....	2017
show port-security oui interface .....	2018
show port-security sticky interface .....	2019
show process cpu .....	2020
show process info .....	2022
show process memory .....	2024
show prom-access .....	2026
show protected-ports.....	2027
show qos flowcontrol interface .....	2028
show qos interface .....	2030
show qos maps .....	2032
show qos maps dscp-cos .....	2033
show qos maps dscp-mutation .....	2034
show qos maps dscp-traffic-class .....	2035
show qos queue interface .....	2036
show qos rcv-queue interface .....	2037
show qos rcv-queue multicast .....	2039
show qos red profiles .....	2040
show qos red statistics interface .....	2042
show qos tx-queue interface .....	2043
show radius servers.....	2045
show rbridge-id .....	2046
show rbridge-running config .....	2047
show rbridge-local-running-config .....	2048
show redundancy .....	2051
show rmon .....	2052
show rmon history .....	2054
show route-map .....	2055
show route-map interface .....	2057
show running reserved-vlan .....	2059
show running-config .....	2060
show running-config aaa .....	2062
show running-config aaa accounting .....	2063
show running-config ag .....	2064
show running-config banner .....	2066
show running-config cee-map .....	2067
show running-config class-map .....	2069
show running-config diag post .....	2070
show running-config dot1x .....	2071
show running-config dpod .....	2072
show running-config event-handler.....	2074
show running-config fabric route mcast .....	2076



show running-config fcoe .....	2077
show running-config fcsp auth .....	2078
show running-config hardware.....	2079
show running-config interface fcoe .....	2083
show running-config interface FibreChannel .....	2084
show running-config interface fortygigabitethernet .....	2086
show running-config interface fortygigabitethernet bpdu-drop .....	2088
show running-config interface fortygigabitethernet cee .....	2089
show running-config interface fortygigabitethernet channel-group .....	2090
show running-config interface fortygigabitethernet description .....	2091
show running-config interface fortygigabitethernet dot1x .....	2092
show running-config interface fortygigabitethernet fabric .....	2094
show running-config interface fortygigabitethernet fcoeport .....	2095
show running-config interface fortygigabitethernet lacp .....	2096
show running-config interface fortygigabitethernet lldp .....	2097
show running-config interface fortygigabitethernet mac .....	2098
show running-config interface fortygigabitethernet mtu .....	2099
show running-config interface fortygigabitethernet port-profile-port .....	2100
show running-config interface fortygigabitethernet priority-tag .....	2101
show running-config interface fortygigabitethernet qos .....	2102
show running-config interface fortygigabitethernet rmon .....	2104
show running-config interface fortygigabitethernet sflow .....	2106
show running-config interface fortygigabitethernet shutdown .....	2107
show running-config interface fortygigabitethernet switchport .....	2108
show running-config interface fortygigabitethernet udld .....	2110
show running-config interface fortygigabitethernet vlan .....	2111
show running-config interface gigabitethernet .....	2112
show running-config interface gigabitethernet bpdu-drop .....	2114
show running-config interface gigabitethernet channel-group .....	2115
show running-config interface gigabitethernet description .....	2116
show running-config interface gigabitethernet dot1x .....	2117
show running-config interface gigabitethernet lacp .....	2119
show running-config interface gigabitethernet lldp .....	2120
show running-config interface gigabitethernet mac .....	2121
show running-config interface gigabitethernet mtu .....	2122
show running-config interface gigabitethernet port-profile-port .....	2123
show running-config interface gigabitethernet priority-tag .....	2124
show running-config interface gigabitethernet qos .....	2125
show running-config interface gigabitethernet rmon .....	2126
show running-config interface gigabitethernet sflow .....	2127
show running-config interface gigabitethernet shutdown .....	2128
show running-config interface gigabitethernet switchport .....	2129
show running-config interface gigabitethernet udld .....	2131
show running-config interface gigabitethernet vlan .....	2132
show running-config interface management .....	2133
show running-config interface port-channel .....	2136
show running-config interface tengigabitethernet .....	2137
show running-config interface tengigabitethernet bpdu-drop .....	2140
show running-config interface tengigabitethernet cee .....	2141
show running-config interface tengigabitethernet channel-group .....	2142

show running-config interface tengigabitethernet description .....	2143
show running-config interface tengigabitethernet dot1x .....	2144
show running-config interface tengigabitethernet fabric .....	2146
show running-config interface tengigabitethernet fcoeport .....	2147
show running-config interface tengigabitethernet lacp .....	2148
show running-config interface tengigabitethernet lldp .....	2149
show running-config interface tengigabitethernet mac .....	2150
show running-config interface tengigabitethernet mtu .....	2151
show running-config interface tengigabitethernet port-profile-port .....	2152
show running-config interface tengigabitethernet priority-tag .....	2153
show running-config interface tengigabitethernet qos .....	2154
show running-config interface tengigabitethernet rmon .....	2155
show running-config interface tengigabitethernet sflow .....	2157
show running-config interface tengigabitethernet shutdown .....	2158
show running-config interface tengigabitethernet switchport .....	2159
show running-config interface tengigabitethernet udd .....	2161
show running-config interface tengigabitethernet vlan .....	2162
show running-config interface vlan .....	2163
show running-config interface vlan ip .....	2164
show running-config ip access-list .....	2166
show running-config ipv6 access-list .....	2167
show running-config ip dns .....	2168
show running-config ip igmp .....	2169
show running-config ip route .....	2170
show running-config ldap-server .....	2171
show running-config line .....	2172
show running-config logging .....	2173
show running-config logging auditlog class .....	2174
show running-config logging raslog .....	2175
show running-config logging syslog-client .....	2176
show running-config logging syslog-facility .....	2177
show running-config logging syslog-server .....	2178
show running-config mac access-list.....	2179
show running-config mac-address-table .....	2180
show running-config monitor .....	2181
show running-config nas server-ip .....	2182
show running-config ntp .....	2183
show running-config ntp authentication-key.....	2184
show running-config openflow-controller.....	2185
show running-config overlay-gateway.....	2186
show running-config ovsdb-server.....	2188
show running-config password-attributes .....	2189
show running-config police-priority-map .....	2191
show running-config policy-map .....	2192
show running-config port-profile .....	2193
show running-config port-profile activate .....	2194
show running-config port-profile fcoe-profile .....	2195
show running-config port-profile qos-profile .....	2196
show running-config port-profile security-profile .....	2197
show running-config port-profile static .....	2198

show running-config port-profile vlan-profile .....	2199
show running-config port-profile-domain .....	2201
show running-config preprovision.....	2202
show running-config protocol cdp .....	2203
show running-config protocol edge .....	2204
show running-config protocol lldp .....	2205
show running-config protocol spanning-tree mstp .....	2207
show running-config protocol spanning-tree pvst .....	2209
show running-config protocol spanning-tree rpvt .....	2210
show running-config protocol spanning-tree rstp .....	2211
show running-config protocol spanning-tree stp .....	2212
show running-config protocol udd .....	2213
show running-config radius-server .....	2214
show running-config rbridge-id .....	2215
show running-config rbridge-id crypto.....	2216
show running-config rbridge-id dhcpd enable.....	2217
show running-config rbridge-id event-handler.....	2218
show running-config rbridge-id hardware-profile.....	2221
show running-config rbridge-id interface .....	2223
show running-config rbridge-id linecard .....	2225
show running-config rbridge-id maps.....	2226
show running-config rbridge-id openflow.....	2228
show running-config rbridge-id ssh .....	2229
show running-config rbridge-id ssh server.....	2230
show running-config rmon .....	2231
show running-config role .....	2232
show running-config route-map .....	2233
show running-config rule .....	2234
show running-config secpolicy .....	2236
show running-config slow .....	2238
show running-config slow-policy .....	2239
show running-config slow-profile .....	2240
show running-config snmp-server .....	2241
show running-config snmp-server context.....	2242
show running-config snmp-server engineid .....	2243
show running-config snmp-server mib community-map.....	2244
show running-config ssh .....	2245
show running-config ssh server .....	2246
show running-config ssh server key-exchange .....	2247
show running-config support autoupload-param .....	2248
show running-config support support-param.....	2249
show running-config switch-attributes .....	2250
show running-config system-monitor .....	2252
show running-config system-monitor-mail .....	2254
show running-config tacacs-server .....	2255
show running-config telnet server .....	2256
show running-config threshold-monitor .....	2257
show running-config threshold-monitor interface .....	2258
show running-config threshold-monitor security .....	2259
show running-config threshold-monitor sfp .....	2260

show running-config username .....	2261
show running-config vcs .....	2263
show running-config vlag-commit-mode.....	2264
show running-config zoning .....	2265
show running-config zoning defined-configuration .....	2266
show running-config zoning enabled-configuration .....	2268
show secpolicy .....	2270
show sflow .....	2272
show sflow-profile .....	2274
show sfm .....	2275
show sfp .....	2277
show slots .....	2278
show span path .....	2280
show spanning-tree .....	2281
show spanning-tree brief .....	2283
show spanning-tree interface .....	2285
show spanning-tree mst brief .....	2287
show spanning-tree mst detail .....	2288
show spanning-tree mst instance .....	2291
show spanning-tree mst interface .....	2293
show ssh server status .....	2295
show ssh server rekey-interval status .....	2297
show startup-config .....	2298
show startup-db .....	2299
show statistics access-list .....	2300
show statistics rpf .....	2304
show storm-control .....	2305
show support .....	2307
show system .....	2308
show system internal arp.....	2309
show system internal asic counter blk.....	2313
show system internal asic counter drop-reason.....	2315
show system internal asic counter interface.....	2316
show system internal asic counter mem blk.....	2318
show system internal bgp evpn interface.....	2320
show system internal bgp evpn l2route type.....	2322
show system internal bgp evpn l3vni.....	2326
show system internal bgp evpn neighbor.....	2328
show system internal bgp evpn routes type.....	2331
show system internal bgp evpn variables.....	2335
show system internal bgp evpn vlan-db.....	2338
show system internal bgp ipv4 config.....	2341
show system internal bgp ipv4 neighbor.....	2345
show system internal bgp ipv4 network.....	2348
show system internal bgp ipv4 nexthop.....	2350
show system internal bgp ipv4 tcpdump.....	2353
show system internal bgp ipv4 variables.....	2356
show system internal bgp ipv6 neighbor.....	2359
show system internal bgp ipv6 network.....	2361
show system internal bgp ipv6 nexthop.....	2363

show system internal bgp ipv6 tcpdump.....	2366
show system internal bgp ipv6 variables.....	2369
show system internal dcm.....	2372
show system internal nas .....	2376
show system internal nsm.....	2377
show system internal nsx.....	2379
show system internal ofagt.....	2381
show system internal ovsdb.....	2384
show system monitor .....	2386
show system pstat interface.....	2388
show telnet server status .....	2391
show threshold monitor .....	2392
show track summary.....	2394
show tunnel .....	2396
show tunnel replicator.....	2399
show tunnel status.....	2401
show udd .....	2403
show udd interface .....	2404
show udd statistics .....	2406
show users .....	2407
show vcs .....	2408
show vcs auto-config-backup.....	2411
show version .....	2412
show version peripheral phy.....	2414
show virtual-fabric status .....	2415
show vlag-partner-info.....	2416
show vlan .....	2417
show vlan brief.....	2419
show vlan classifier .....	2421
show vlan private-vlan .....	2422
show vlan rspan-vlan .....	2423
show vnetwork .....	2424
show vrf .....	2428
show vrrp.....	2431
show zoning enabled-configuration .....	2437
show zoning operation-info .....	2438

<b>Commands shutdown through Z.....</b>	<b>2441</b>
shutdown .....	2441
shutdown (STP).....	2443
shutdown (UDLD).....	2444
shutdown (VXLAN).....	2445
shutdown-time .....	2446
site .....	2448
slot .....	2450
snmp-server community .....	2451
snmp-server contact .....	2453
snmp-server context .....	2454
snmp-server enable trap.....	2455
snmp-server engineid local .....	2456
snmp-server group .....	2457

snmp-server host .....	2459
snmp-server location .....	2461
snmp-server mib community-map.....	2462
snmp-server sys-descr .....	2464
snmp-server three-tuple-if enable.....	2465
snmp-server user .....	2466
snmp-server v3host .....	2469
snmp-server view .....	2471
snmp-server offline-if enable.....	2472
snmp-server trap link-status.....	2473
source .....	2474
span session .....	2476
spanning-tree autoedge .....	2477
spanning-tree bpdu-mac .....	2478
spanning-tree cost .....	2479
spanning-tree edgeport .....	2480
spanning-tree guard root .....	2482
spanning-tree hello-time .....	2484
spanning-tree ieee-bpdu limit-vlan-flood.....	2485
spanning-tree instance .....	2486
spanning-tree link-type .....	2488
spanning-tree peer-switch.....	2489
spanning-tree portfast .....	2490
spanning-tree priority .....	2492
spanning-tree restricted-role .....	2493
spanning-tree restricted-tcn .....	2494
spanning-tree shutdown .....	2495
spanning-tree vlan .....	2496
speed (Ethernet).....	2497
speed (Fibre Channel).....	2499
speed (FlexPort).....	2500
speed (LAG).....	2502
speed (port-channel).....	2503
spt-threshold .....	2504
ssh .....	2505
ssh client cipher.....	2507
ssh client cipher non-cbc.....	2508
ssh client key-exchange .....	2509
ssh client mac.....	2511
ssh server cipher.....	2512
ssh server cipher non-cbc.....	2513
ssh server key-exchange .....	2514
ssh server mac.....	2516
ssh server max-sessions.....	2517
ssh server rekey-interval .....	2519
ssh server shutdown .....	2520
ssh server standby enable.....	2522
static-network .....	2523
storm-control ingress .....	2525
summary-address (OSPFv2).....	2527

summary-address (OSPFv3).....	2529
support autoupload enable .....	2531
support autoupload-param .....	2532
support support-param.....	2533
suppress-arp.....	2534
suppress-nd.....	2535
switch-attributes .....	2536
switchport .....	2538
switchport access .....	2539
switchport mode .....	2541
switchport mode private-vlan .....	2542
switchport mode trunk-no-default-native .....	2544
switchport port-security .....	2545
switchport port-security mac-address .....	2546
switchport port-security max .....	2547
switchport port-security oui .....	2548
switchport port-security shutdown-time .....	2549
switchport port-security sticky .....	2550
switchport port-security violation .....	2551
switchport private-vlan association trunk .....	2552
switchport private-vlan host-association .....	2553
switchport private-vlan mapping .....	2554
switchport private-vlan trunk allowed vlan .....	2555
switchport private-vlan trunk native-vlan .....	2557
switchport trunk allowed vlan rspan-vlan .....	2558
switchport trunk default-vlan .....	2560
switchport trunk native-vlan .....	2561
switchport trunk native-vlan-untagged .....	2563
switchport trunk native-vlan-xtagged .....	2564
switchport trunk tag native-vlan .....	2566
system tunnel suppress-debounce.....	2567
system-description .....	2568
system-id oui.....	2569
system-mode maintenance.....	2571
system-monitor .....	2572
system-monitor-mail .....	2575
system-name .....	2577
table-map .....	2578
tacacs-server .....	2580
tcp burstrate .....	2583
telnet .....	2584
telnet server shutdown .....	2586
telnet server standby enable.....	2588
terminal .....	2589
threshold-monitor cpu .....	2591
threshold-monitor interface .....	2593
threshold-monitor memory .....	2596
threshold-monitor security .....	2598
threshold-monitor sfp .....	2601
timeout fnm .....	2604

timers .....	2605
timers (BGP).....	2607
timers (OSPFv3).....	2608
traceroute .....	2610
track (Fabric-Virtual-Gateway).....	2612
track (LST).....	2614
track (VRRP).....	2616
transmit-holdcount .....	2618
transport-service .....	2619
trigger.....	2620
trigger-function.....	2622
trigger-mode.....	2624
trunk-enable .....	2626
tunable-optics.....	2627
tunnel replicator bum-vlans redistribute.....	2631
tunnel tagged-ieee-bpdu .....	2632
tunnel-traceroute.....	2633
type .....	2636
type (FlexPort).....	2638
udld enable .....	2639
unhide built-in-self-test.....	2640
unhide fips .....	2641
unlock username .....	2642
update-time .....	2643
uplink-switch enable.....	2645
usb .....	2646
usb dir .....	2647
usb remove .....	2648
user (alias configuration).....	2649
username .....	2650
username admin enable false .....	2652
username user enable false .....	2653
use-v2-checksum.....	2654
vcenter .....	2655
vcenter discovery (ignore delete responses).....	2657
vcenter vlan-create.....	2658
vcs auto-config-backup timer.....	2660
vcs config snapshot .....	2662
vcs logical-chassis enable default-config.....	2664
vcs replace rbridge-id .....	2665
vcs set-rbridge-id.....	2666
vcs vcsid .....	2667
vcs virtual .....	2668
vcs virtual-fabric enable .....	2670
version.....	2672
virtual-fabric .....	2673
virtual-ip .....	2674
virtual-mac .....	2676
vlag ignore-split .....	2677
vlag-commit-mode disable.....	2678



vlan classifier activate group .....	2679
vlan classifier group .....	2680
vlan classifier rule .....	2681
vlan dot1q tag native .....	2683
vlan-profile (AMPP).....	2684
vnetwork reconcile vcenter.....	2685
vnetwork vcenter discover .....	2686
vni (EVPN).....	2687
vni (VRF).....	2688
vni add .....	2690
vni remove.....	2692
vrf .....	2693
vrf forwarding.....	2694
vrf-lite-capability .....	2695
vrf mgmt-vrf.....	2696
vrrp-acceptmode-disable.....	2697
vrrp-extended-group .....	2698
vrrp-group .....	2699
vtep-discovery .....	2701
write erase.....	2702
zoning defined-configuration alias .....	2704
zoning defined-configuration cfg .....	2706
zoning defined-configuration zone .....	2708
zoning enabled-configuration cfg-action cfg-clear .....	2710
zoning enabled-configuration cfg-action cfg-save .....	2712
zoning enabled-configuration cfg-action cfg-transaction-abort .....	2713
zoning enabled-configuration cfg-name .....	2714
zoning enabled-configuration default-zone-access .....	2716



# Preface

---

- Document conventions..... 39
- Extreme resources..... 40
- Document feedback..... 40
- Contacting Extreme Technical Support..... 41

## Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Extreme technical documentation.

## Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

### NOTE

A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

### ATTENTION

An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.



### CAUTION

A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.



### DANGER

A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

## Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used to highlight specific words or phrases.

Format	Description
<b>bold text</b>	Identifies command names. Identifies keywords and operands. Identifies the names of GUI elements.
<i>italic text</i>	Identifies text to enter in the GUI. Identifies emphasis. Identifies variables.
Courier font	Identifies document titles. Identifies CLI output.

Format	Description
	Identifies command syntax examples.

## Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
<b>bold text</b>	Identifies command names, keywords, and command options.
<i>italic text</i>	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional.  Default responses to system prompts are enclosed in square brackets.
{ x   y   z }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x   y	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, <i>member[member...]</i> .
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

## Extreme resources

Visit the Extreme website to locate related documentation for your product and additional Extreme resources.

White papers, data sheets, and the most recent versions of Extreme software and hardware manuals are available at [www.extremenetworks.com](http://www.extremenetworks.com). Product documentation for all supported releases is available to registered users at [www.extremenetworks.com/support/documentation](http://www.extremenetworks.com/support/documentation).

## Document feedback

Quality is our first concern at Extreme, and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you.

You can provide feedback in two ways:

- Use our short online feedback form at <http://www.extremenetworks.com/documentation-feedback-pdf/>
- Email us at [internalinfodev@extremenetworks.com](mailto:internalinfodev@extremenetworks.com)

Provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

# Contacting Extreme Technical Support

As an Extreme customer, you can contact Extreme Technical Support using one of the following methods: 24x7 online or by telephone. OEM customers should contact their OEM/solution provider.

If you require assistance, contact Extreme Networks using one of the following methods:

- [GTAC \(Global Technical Assistance Center\)](#) for immediate support
  - Phone: 1-800-998-2408 (toll-free in U.S. and Canada) or +1 408-579-2826. For the support phone number in your country, visit: [www.extremenetworks.com/support/contact](http://www.extremenetworks.com/support/contact).
  - Email: [support@extremenetworks.com](mailto:support@extremenetworks.com). To expedite your message, enter the product name or model number in the subject line.
- [GTAC Knowledge](#) - Get on-demand and tested resolutions from the GTAC Knowledgebase, or create a help case if you need more guidance.
- [The Hub](#) - A forum for Extreme customers to connect with one another, get questions answered, share ideas and feedback, and get problems solved. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.
- [Support Portal](#) - Manage cases, downloads, service contracts, product licensing, and training and certifications.

Before contacting Extreme Networks for technical support, have the following information ready:

- Your Extreme Networks service contract number and/or serial numbers for all involved Extreme Networks products
- A description of the failure
- A description of any action(s) already taken to resolve the problem
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load at the time of trouble (if known)
- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any related RMA (Return Material Authorization) numbers



# About This Document

---

- Supported hardware and software.....43
- What's new in this document.....43

## Supported hardware and software

In those instances in which procedures or parts of procedures documented here apply to some devices but not to others, this guide identifies exactly which devices are supported and which are not.

Although many different software and hardware configurations are tested and supported by Extreme Networks, Inc. for Network OS, documenting all possible configurations and scenarios is beyond the scope of this document.

The following hardware platforms are supported by this release of Network OS:

- ExtremeSwitching VDX 2746
- ExtremeSwitching VDX 6740
  - ExtremeSwitching VDX 6740-48
  - ExtremeSwitching VDX 6740-64
- ExtremeSwitching VDX 6740T
  - ExtremeSwitching VDX 6740T-48
  - ExtremeSwitching VDX 6740T-64
  - ExtremeSwitching VDX 6740T-1G
- ExtremeSwitching VDX 6940-36Q
- ExtremeSwitching VDX 6940-144S
- ExtremeSwitching VDX 8770
  - ExtremeSwitching VDX 8770-4
  - ExtremeSwitching VDX 8770-8

To obtain information about a Network OS version other than this release, refer to the documentation specific to that version.

## What's new in this document

The following command changes have been made in Network OS 7.2.0c:

### New commands

- **bfd-shutdown**

### Modified commands

- None

## Deprecated commands

- None

The following command changes have been made in Network OS 7.2.0:

## New commands

The following commands have been added in this release.

- `advertise bgp-auto-nbr-tlv`
- `graceful-shutdown`
- `neighbor accept-lldp-neighbors`
- `neighbor graceful-shutdown`
- `protected-port enable`
- `rpf-mode`
- `show ip pim-sparse`
- `show protected-ports`
- `show statistics rpf`
- `tunnel tagged-ieee-bpdu`
- `tunnel traceroute`
- `uplink-switch enable`
- `vcenter vlan-create`

## Modified commands

The following commands have been modified in this release.

- `debug show bp-stats interface`
- `hardware-profile`
- `match community`
- `rule (maps)`
- `show hardware profile`
- `show ip bgp neighbors`
- `show ip bgp neighbors`
- `show ip interface`
- `show ip route`
- `show ipv6 route`
- `show lldp`
- `show lldp neighbors`
- `show mac-address-table`
- `show maps dashboard`
- `show vlan brief`



## Deprecated commands

The following commands have been deprecated in this release.

- None



# Using the Network OS CLI

---

• DCB command line interface.....	47
• RBAC permissions .....	47
• Default roles.....	47
• Accessing the Network OS CLI through Telnet .....	48
• Network OS CLI command modes.....	48
• CLI keyboard shortcuts.....	54
• Using the do command as a shortcut.....	55
• Displaying CLI commands and command syntax.....	55
• Completing CLI commands.....	56
• Using CLI command output modifiers.....	56
• Considerations for show command output .....	57
• User-configurable VLAN IDs.....	57
• Debug and system diagnostic commands.....	57

## DCB command line interface

The Data Center Bridging (DCB) CLI is designed to support the management of DCB and Layer 2 Ethernet switching functionality. The Network OS CLI uses an industry-standard hierarchical shell familiar to Ethernet/IP networking administrators.

The system starts up with the default Network OS configuration and the DCB startup configuration. After logging in, you are in the Network OS shell. For information on accessing the DCB commands from the Network OS shell, refer to [Network OS CLI command modes](#) on page 48.

## RBAC permissions

Role-Based Access Control (RBAC) defines the capabilities that a user account has based on the role the account has been assigned.

A role is an entity that defines the access privileges of the user accounts on the switch. A user is associated with one role. For more information, refer to the "User Accounts and Passwords" section of the *Extreme Network OS Security Configuration Guide*.

## Default roles

Attributes of default roles cannot be modified; however, the default roles can be assigned to non-default user accounts. The following roles are default roles:

- The admin role has the highest privileges. All CLIs are accessible to the user associated with the admin role. By default, the admin role has read and write access.
- The user role has limited privileges that are mostly restricted to show commands in privileged EXEC mode. User accounts associated with the user role cannot access configuration CLIs that are in global configuration mode. By default, the user role has read-only access.

# Accessing the Network OS CLI through Telnet

**NOTE**

While this example uses the admin role to log in to the switch, both the admin and the user role can be used.

The procedure to access the Network OS CLI is the same through either the console interface or through a Telnet session; both access methods bring you to the login prompt.

```
switch login: admin
Password:*****
switch#
```

**NOTE**

Multiple users can open Telnet sessions and issue commands by using privileged EXEC mode. Network OS supports up to 32 Telnet sessions with the admin login.

## Network OS CLI command modes

The following lists the major Network OS CLI command modes and describes how to access them.

**NOTE**

Use the **pwd** command to view the mode of the current working directory. This command functions in global configuration mode and the modes accessed from global configuration mode.

**NOTE**

Pressing Ctrl+Z or entering the end command in any mode returns you to privileged EXEC mode. Entering exit in any mode returns you to the previous mode.

**TABLE 1** Network OS CLI command modes

Command mode	Prompt	How to access the command mode	Description
Privileged EXEC mode	device#	This is the default mode for the switch.	Display and change system parameters. Note that this is the administrative mode and includes the basic configuration commands.
Global configuration mode	device(config)#	From privileged EXEC mode, enter the <b>configure terminal</b> command.	Configure features that affect the entire switch.
line vty configuration mode	device(config)#line vty exec-timeout 60 device(config-line-vty)#	From global configuration mode, enter the <b>line vty</b> command.	Specify the amount of time a CLI session can be idle before it logs you out.
RBridge ID configuration mode	RBridge ID:  device(config)# rbridge-id 1 device(config-rbridge-id-1)#	From global configuration mode, specify a node by entering the <b>rbridge-id rbridge_id</b> command, where <i>rbridge_id</i> is the RBridge ID of the selected node.	Configure features and issue show commands specific to an individual node in a Virtual Cluster Switching (VCS) environment.
Interface subtype configuration mode	Port-channel:  device(config)# port-channel 63 device(config-Port-channel-63)#	From global configuration mode, specify an interface by entering one of the following commands:  <ul style="list-style-type: none"> <li>• <b>interface fcoe</b></li> <li>• <b>interface fortygigabitethernet</b></li> </ul>	Access and configure individual interface subtypes.  Enter ? at a command prompt to see what interface subtypes are available for that command.

TABLE 1 Network OS CLI command modes (continued)

Command mode	Prompt	How to access the command mode	Description
	<p>10-Gigabit Ethernet (DCB port):</p> <pre>device(config)# interface tengigabitethernet 1/0/1 device(config-if-te-1/0/1)#</pre> <p>VLAN:</p> <pre>device(config)# interface vlan 1 device(config-Vlan-1)#</pre> <p>VE (global):</p> <pre>device(config)# int ve 56 device(config-Ve-56)#</pre> <p>VE (RBridge):</p> <pre>device(config)# rbridge-id 11 device(config-rbridge-id-11)# int ve 56 device(config-rbridge-Ve-56)#</pre> <p>Management:</p> <pre>device(config)# interface management 1/3/1 device(config-Management-1/0/1)</pre> <p>Loopback:</p> <pre>device(config)# interface loopback 1/3/1 device(config-Loopback-1/0/1)</pre>	<ul style="list-style-type: none"> <li>• <b>interface gigabitethernet</b></li> <li>• <b>interface fibrechannel</b></li> <li>• <b>interface hundredgigabitethernet</b></li> <li>• <b>interface loopback</b></li> <li>• <b>interface management</b></li> <li>• <b>interface port-channel</b></li> <li>• <b>interface tengigabitethernet</b></li> <li>• <b>interface ve</b></li> <li>• <b>interface vlan</b></li> </ul>	
Protocol configuration mode	<p>LLDP:</p> <pre>device(config-lldp)#</pre> <p>Spanning tree:</p> <pre>device(config-mstp)# device(config-rstp)# device(config-stp)# device(config-pvst)# device(config-rpvst)# device(config-udld)#</pre>	<p>From global configuration mode, specify a protocol by entering one of the following commands:</p> <ul style="list-style-type: none"> <li>• <b>protocol lldp</b></li> <li>• <b>protocol spanning-tree mstp</b></li> <li>• <b>protocol spanning-tree rstp</b></li> <li>• <b>protocol spanning-tree stp</b></li> <li>• <b>protocol spanning-tree pvst</b></li> <li>• <b>protocol spanning-tree rapid-pvst</b></li> <li>• <b>protocol udld</b></li> </ul>	Access and configure protocols.
FCoE configuration mode	<p>FCoE:</p> <pre>device(config-fcoe)#</pre> <p>FCoE fabric-map sub-mode:</p> <pre>device(config-fcoe-fabric-map)#</pre> <p>FCoE map sub-mode:</p> <pre>device(config-fcoe-map)#</pre>	<p>From global configuration mode, use the <b>fcoe</b> command to enter FCoE configuration mode.</p> <p>From FCoE configuration mode, specify an FCoE sub-mode by entering one of the following commands:</p> <ul style="list-style-type: none"> <li>• <b>fabric-map</b></li> <li>• <b>fcoe-map</b></li> </ul>	Access and configure FCoE features.

TABLE 1 Network OS CLI command modes (continued)

Command mode	Prompt	How to access the command mode	Description
Access Gateway (AG) configuration mode	AG configuration mode: <pre>device (config-rbridge-12-ag) #</pre> N_Port configuration mode: <pre>device (config-rbridge-12-ag-nport-if-fi-port_num) #</pre> Port Group configuration mode: <pre>device (config-rbridge-12-ag-pg-pg_id) #</pre>	From RBridge-ID configuration mode, enter the <b>ag</b> command.  From AG or PG configuration mode, enter the <b>nport interface fiberchannel port_num</b> command.  From AG configuration mode, enter <b>pg pg_id</b> where <i>pg_id</i> is the port group identification number.	Access and configure Access Gateway features.
AMPP port-profile configuration mode	AMPP port-profile: <pre>device (config-port-profile-name) #</pre> VLAN-profile sub-mode: <pre>device (config-vlan-profile) #</pre> QoS-profile sub-mode: <pre>device (config-qos-profile) #</pre> FCoE-profile sub-mode: <pre>device (config-fcoe-profile) #</pre> Security-profile sub-mode: <pre>device (config-security-profile) #</pre>	From the global configuration mode, enter the <b>port-profile</b> command to enter port-profile configuration mode.  From port-profile configuration mode, specify an AMPP sub-mode by entering one of the following commands: <ul style="list-style-type: none"> <li>• <b>vlan-profile</b></li> <li>• <b>qos-profile</b></li> <li>• <b>fcoe-profile</b></li> <li>• <b>security-profile</b></li> </ul>	Access and configure AMPP features.
Routing protocol configuration mode	BGP: <pre>device (config) #</pre> <pre>device (config-rbridge-id-1) #</pre> <pre>device (config-bgp-router) #</pre> BGP route-map configuration mode: <pre>device (config-rbridge-id-1) #</pre> <pre>device (config-route-map-myroutemap/permit/1) #</pre> BGP address-family IPv4 unicast mode: <pre>device (config-bgp-router) #</pre> <pre>device (config-bgp-ipv4u) #</pre> BGP address-family IPv4 unicast VRF mode: <pre>device (config-bgp-router) #</pre> <pre>device (config-bgp-ipv4u-vrf) #</pre> BGP address-family IPv6 unicast mode: <pre>device (config-bgp-router) #</pre> <pre>device (config-bgp-ipv6u) #</pre> BGP address-family IPv6 unicast VRF mode: <pre>device (config-bgp-router) #</pre> <pre>device (config-bgp-ipv6u-vrf) #</pre>	From global configuration mode, specify an RBridge ID to enter RBridge ID configuration mode.  From RBridge ID configuration mode, use the <b>router bgp</b> command to enter BGP configuration mode.  From RBridge ID configuration mode, use the <b>route-map</b> command with a <b>permit</b> or <b>deny</b> statement and an instance number to enter BGP route-map configuration mode.  From BGP configuration mode, use the <b>address-family ipv4 unicast</b> command to enter BGP address-family IPv4 unicast configuration mode.  From BGP configuration mode, use the <b>address-family ipv4 unicast</b> command with the <b>vrf</b> keyword and specify a VRF name to enter BGP address-family IPv4 unicast VRF configuration mode.  From BGP configuration mode, use the <b>address-family ipv6 unicast</b> command to enter BGP address-	Configure Border Gateway Protocol routing protocol

TABLE 1 Network OS CLI command modes (continued)

Command mode	Prompt	How to access the command mode	Description
		family IPv6 unicast configuration mode.  From BGP configuration mode, use the <b>address-family ipv6 unicast</b> command with the <b>vrf</b> keyword and specify a VRF name to enter BGP address-family IPv6 unicast VRF configuration mode.	
	OSPF router:  <pre>device (config) # device (config-rbridge-id-5) # device (config-router-ospf-vrf-default-vrf) #</pre> OSPFv3 router:  <pre>device (config) # device (config-rbridge-id-5) # device (config-ipv6-router-ospf-vrf-default-vrf) #</pre> OSPF router VRF:  <pre>device (config) # device (config-rbridge-id-5) # device (config-router-ospf-vrf-red) #</pre> OSPFv3 router VRF:  <pre>device (config) # device (config-rbridge-id-5) # device (config-ipv6-router-ospf-vrf-red) #</pre>	From RBridge ID configuration mode, use the <b>router ospf</b> command to enter router OSPF configuration mode.  From RBridge ID configuration mode, use the <b>router ospf</b> command with the <b>vrf</b> keyword and specify a VRF to enter router OSPF VRF configuration mode.  From RBridge ID configuration mode, use the <b>ipv6 router ospf</b> command to enter router OSPFv3 configuration mode.  From RBridge ID configuration mode, use the <b>ipv6 router ospf</b> command with the <b>vrf</b> keyword and specify a VRF to enter router OSPFv3 VRF configuration mode.	Configure Open Short Path First or Open Shortest Path First version 3 routing protocol
	PIM:  <pre>device (config) # device (config-rbridge-id-5) # device (config-pim-router) #</pre>	From Bridge ID configuration mode, use the <b>router pim</b> command to enter PIM configuration mode.	Configure Protocol Independent Multicast routing protocol
Virtual-router-group configuration mode	<pre>device (config) # rbridge-id 101 device (config-rbridge-id-101) # int ve 25 device (config-ve-25) # vrrp-extended-group 1 device (config-vrrp-extended-group-1) #</pre>	From RBridge ID configuration mode, use the <b>int ve</b> command to enter VE configuration mode. Then use the <b>vrrp-extended-group</b> command to enter virtual-router-group configuration mode.	
ACL configuration mode	Standard ACL:  <pre>device (config-macl-std) # device (config-ipacl-std) # device (config-ip6acl-std) #</pre> Extended ACL:  <pre>device (config-macl-ext) # device (config-ipacl-ext) # device (config-ip6acl-ext) #</pre>	From global configuration mode, enter one of the following commands: <ul style="list-style-type: none"> <li>• <b>mac access-list standard</b></li> <li>• <b>mac access-list extended</b></li> <li>• <b>ip access-list standard</b></li> <li>• <b>ip access-list extended</b></li> <li>• <b>ipv6 access-list standard</b></li> <li>• <b>ipv6 access-list extended</b></li> </ul>	Create permit/deny access rules within access control lists (ACLs).

TABLE 1 Network OS CLI command modes (continued)

Command mode	Prompt	How to access the command mode	Description
ARP access-list configuration mode	<code>device (config-arp-acl) #</code>	From global configuration mode, enter the <b>arp access-list</b> <i>name</i> command.	For dynamic ARP inspection (DAI), define rules that specify which IP/MAC address bindings are trusted.
CEE map configuration mode	<code>device (config-cee-map-default) #</code>	From global configuration mode, enter the <b>cee-map default</b> command.	Access and configure CEE map features.
ELD configuration mode	<code>device (config) # protocol edge-loop-detection</code> <code>device (config-eld) #</code>	From global configuration mode, enter the <b>protocol edge-loop-detection</b> command.	Configure edge loop detection.
Hardware configuration mode	<code>device (config) # hardware</code> <code>device (config-hardware) #</code>	From global configuration mode, enter the <b>hardware</b> command.	This mode is a prerequisite for entering connector and port-group mode.
Connector configuration mode	<code>device # hardware</code> <code>device (config-hardware) # connector</code> <code>device (config-connector [n]/n/n) #</code>	From hardware configuration mode, specify the connector node and [ <i>rbridge-id</i> ]/ <i>slot</i> / <i>port</i> information.	Connector mode is used to enable breakout on ports. When breakout is enabled, ports are appended in the output with a colon(:) followed by values 1-4.
DSCP mutation mapping configuration mode	<code>device (dscp-mutation-mapname) #</code>	From global configuration mode, remap incoming DSCP values by entering the <b>qos map dscp-mutation</b> <i>mapname</i> command:	
DSCP-to-CoS priority mapping configuration mode	<code>device (dscp-cos-mapname) #</code>	From global configuration mode, create a DSCP to CoS priority map by entering the <b>qos map dscp-cos</b> <i>mapname</i> command.	
DSCP-to-traffic-class mapping configuration mode	<code>device (dscp-traffic-class-mapname) #</code>	From global configuration mode, create a DSCP to traffic class map by entering the <b>qos map dscp-traffic-class</b> <i>mapname</i> command:	
Port-group configuration mode	<code>device (config-port-group-1/3/9) #</code>	From hardware configuration mode, enter the <b>port-group</b> command followed by a port group identification:  <b>port-group</b> <i>rbridge-id</i> / <i>slot</i> / <i>port-group-id</i>  The port-group-id is specific to the VDX 8770 device 27x40 GbE line card.	This mode allows you to enable Performance or Density operating modes on a specific port group on the 27x40 GbE line card only.
QoS Policer configuration mode	Police Priority Map: <code>device (config-policemap) #</code>  Class Map: <code>device (config-classmap) #</code>  Policy Map: <code>device (config-policymap) #</code>  Policy-class-map submode: <code>device (config-policymap-class) #</code>	From global configuration mode, specify a Policer configuration mode by entering one of these command:  <ul style="list-style-type: none"> <li><b>police-priority-map</b> <i>mapname</i></li> <li><b>class-map</b> <i>mapname</i></li> <li><b>policy-map</b> <i>mapname</i></li> </ul> To enter the policy-class-map submode from policy-map mode, enter <b>class</b> <i>classmapname</i>  To enter the policy-class-map-policer attributes sub-mode from policy-	



TABLE 1 Network OS CLI command modes (continued)

Command mode	Prompt	How to access the command mode	Description
	Policy-class-map-policer attributes submode:  device (config-policymap-class-police) #	map-class mode, enter <b>police</b> followed by the policing attributes.	
Alias configuration mode	device (config-alias-config) #	From global configuration mode, enter the <b>alias-config</b> command.  Use the <b>alias string</b> expansion command to create aliases.	Access configure alias features.
User alias configuration mode	device (config-alias-config-user) #	From alias configuration mode, enter the <b>user name</b> command.	Access configure user alias features.
Policymap configuration mode	device (config-policymap) #	From global configuration mode, enter the <b>policy-map name</b> command.	
Policymap class map configuration mode	device (config-policymap-class) #	From policymap configuration mode, enter the <b>class name</b> command.	
Policymap class police configuration mode	device (config-policymap-class-police) #	From policymap class configuration mode, enter the <b>police cir value</b> command.	
VRF configuration mode	(config-rbridge-12-vrf-vrf_name) #	From RBridge ID configuration mode, enter the <b>vrf name</b> command.	
Python command shell	>>>	From privileged EXEC mode, enter the <b>python</b> command.	Launch a Python script or access a Python command shell.
Event-handler configuration mode	device (config-event-handler-eH1) #	From global configuration mode, enter the <b>event-handler name</b> command.	Access an event-handler profile, which can execute a Python script when a specified trigger occurs.
Event-handler activation mode	device (config-activate-eH1) #	From RBridge ID configuration mode, enter the <b>event-handler activate name</b> command.	Activate an event handler on an RBridge, with the option of defining advanced configuration commands.
OpenFlow configuration mode	device (config) # openflow-controller mycontroller	From global configuration mode, enter the <b>open-controller name</b> command.	
	device (config-rbridge-id-1) # openflow logical-instance 1 device (config-logical-instance 1) # activate	From RBridge ID configuration mode, enter the <b>openflow logical instance 1</b> command and then the <b>activate</b> command to activate the logical instance.	
	device (config) # interface tengigabitethernet 12/0/12 device (conf-if-te-12/0/12) # lldp disable device (conf-if-te-12/0/12) # openflow logical-instance 1 device (conf-if-te-12/0/12) # openflow enable	From interface subtype configuration mode, enter the <b>openflow logical instance 1</b> command and then the <b>openflow enable</b> command.	

**TABLE 1** Network OS CLI command modes (continued)

Command mode	Prompt	How to access the command mode	Description
Overlay gateway configuration mode	device# configure terminal device(config)# overlay-gateway gateway1 device(config-overlay-gw-gateway1)#	From global configuration mode, enter the <b>overlay-gateway name</b> command.	
BGP address-family L2VPN EVPN configuration mode	device# configure terminal device(config)# rbridge-id 10 device(config-rbridge-id-10)# device(config-rbridge-id-10)# router bgp device(config-bgp-router)# address-family l2vpn evpn device(config-bgp-evpn)#	From global configuration mode, enter RBridge ID configuration mode, then enter BGP router configuration mode, and then enter the <b>address-family l2vpn evpn</b> command.	
EVPN instance configuration mode	device# configure terminal device(config)# rbridge-id 10 device(config-rbridge-id-10)# device(config-rbridge-id-10)# evpn-instance myinstance device(config-evpn-instance-myinstance)#	From global configuration mode, enter RBridge ID configuration mode, and then enter the <b>evpn-instance name</b> command.	
OVSDB server configuration mode	device# ovdsb-server myserver device(config-ovdsb-server-myserver)#	From global configuration mode, enter the <b>ovdsb-server name</b> command to specify an SSL server and enter OVSDB server configuration mode	

## CLI keyboard shortcuts

The following table lists CLI keyboard shortcuts.

**TABLE 2** Network OS CLI keyboard shortcuts

Keystroke	Description
<b>Ctrl+B</b> (or the left arrow key)	Moves the cursor back one character.
<b>Ctrl+F</b> (or the right arrow key)	Moves the cursor forward one character.
<b>Ctrl+A</b>	Moves the cursor to the beginning of the command line.
<b>Ctrl+E</b>	Moves the cursor to the end of the command line.
<b>Esc B</b>	Moves the cursor back one word.
<b>Esc F</b>	Moves the cursor forward one word.
<b>Ctrl+Z</b>	Returns to privileged EXEC mode. Using Ctrl+Z in privileged EXEC mode executes partial commands.
<b>Ctrl+P</b> (or the up arrow key)	Displays commands in the history buffer with the most recent command displayed first.
<b>Ctrl+N</b> (or the down arrow key)	Displays commands in the history buffer with the most recent command displayed last.

### NOTE

In privileged EXEC mode, use the **show history** command to list the commands most recently entered. The device retains the history of the last 1000 commands entered for the current session.

## Using the do command as a shortcut

You can use the **do** command to save time when you are working in any configuration mode and you want to run a command in privileged EXEC mode.

For example, if you are configuring LLDP and you want to execute a privileged EXEC mode command, such as the **dir** command, you would first have to exit the LLDP configuration mode. By using the **do** command with the **dir** command, you can ignore the need to change configuration modes, as shown in the following example.

```
device(conf-lldp)# do dir
Contents of flash://
-rw-r----- 1276 Wed Feb 4 07:08:49 2009 startup_rmon_config
-rw-r----- 1276 Wed Feb 4 07:10:30 2009 rmon_config
-rw-r----- 1276 Wed Feb 4 07:12:33 2009 rmon_configuration
-rw-r----- 1276 Wed Feb 4 10:48:59 2009 startup-config
```

## Displaying CLI commands and command syntax

You can display commands and syntax information in any mode and from any point in the command hierarchy.

Enter a question mark (?) in any command mode to display the list of commands available in that mode.

```
device(conf-lldp)# ?

Possible completions:
advertise      The Advertise TLV configuration.
description    The User description
disable        Disable LLDP
do             Run an operational-mode command
exit           Exit from current mode
hello          The Hello Transmit interval.
help           Provide help information
iscsi-priority Configure the Ethernet priority to advertise for iSCSI
mode           The LLDP mode.
multiplier     The Timeout Multiplier
no             Negate a command or set its defaults
profile        The LLDP Profile table.
pwd            Display current mode path
system-description The System Description.
system-name    The System Name
top            Exit to top level and optionally run command
```

To display a list of commands that start with the same characters, type the characters followed by the question mark (?).

```
device# e?

Possible completions:
exit  Exit the management session
```

To display the keywords and arguments associated with a command, enter the keyword followed by the question mark (?).

```
device# terminal ?
Possible completions:
length  Sets Terminal Length for this session
monitor Enables terminal monitoring for this session
no      Sets Terminal Length for this session to default :24.
timeout Sets the interval that the EXEC command interpreter wait for user input.
```

If the question mark (?) is typed within an incomplete keyword, but the keyword matches several keywords, the CLI displays help for all the matching keywords.

```
device# show d?
```

```
Possible completions:
debug  Debug
diag   Show diag related information
dot1x  802.1x configuration
dpod   Provides DPOD license information.
```

The Network OS CLI accepts abbreviations for commands. This example is the abbreviation for the **show qos interface all** command.

```
device# sh q i a
```

If the device does not recognize a command after you press **Enter**, an error message displays.

```
device# hookup
          ^
syntax error: unknown argument.
```

If you enter an incomplete command, an error message displays.

```
device# show
          ^
syntax error: unknown argument.
```

## Completing CLI commands

To complete the spelling of commands or keywords automatically, begin typing the command or keyword and then press **Tab**. For example, at the CLI command prompt, type `te` and press **Tab**:

```
device# te
```

The CLI displays the following command.

```
device# terminal
```

If there is more than one command or keyword associated with the characters typed, the CLI displays all choices. For example, at the CLI command prompt, type `show l` and press **Tab**.

```
device# show l
```

The CLI displays the following command.

```
Possible completions:
lACP      LACP commands
license   Display license keys installed on the switch.
lldp      Link Layer Discovery Protocol (LLDP).
logging   Show logging
```

## Using CLI command output modifiers

You can filter the output of the CLI **show** commands by using the output modifiers described below.

**TABLE 3** CLI command output modifiers

Output modifier	Description
<b>append</b>	Appends the output to a file.
<b>redirect</b>	Redirects the command output to the specified file.
<b>include</b>	Displays the command output that includes the specified expression.
<b>exclude</b>	Displays the command output that excludes the specified expression.
<b>begin</b>	Displays the command output that begins with the specified expression.

TABLE 3 CLI command output modifiers (continued)

Output modifier	Description
<b>last</b>	Displays only the last few lines of the command output.
<b>tee</b>	Redirects the command output to the specified file. Notice that this modifier also displays the command output.
<b>until <i>string</i></b>	Ends the output when the output text matches the string.
<b>count</b>	Counts the number of lines in the output.
<b>linnum</b>	Enumerates the lines in the output.
<b>more</b>	Paginates the output.
<b>nomore</b>	Suppresses the pagination of the output.
<b>FLASH</b>	Redirects the output to flash memory.

## Considerations for show command output

Network OS contains many versions of the **show** command. The output of the **show** command changes depending on your configuration and situation. However, in general terms the **show** command falls into one of two categories:

- Any **show** commands that are fabric (global configuration) in nature, such as VLAN, MAC Address table, AMPP, Zoning, and so on, should display or clear the information for all nodes in a logical chassis.
- Any **show** commands that are local to a switch, such as Layer 3 or Layer 2 functionality (for example, sFlow, SPAN, and so on), should display the local information by default, and display different switch information specific to an RBridge ID.

## User-configurable VLAN IDs

On the VDX family of devices, VLANs are treated as interfaces from a configuration point of view. By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). VLAN 1002 is reserved for FCoE VLAN functionality on all VDX switches. Valid VLAN IDs (those configurable by the user), as well as VLAN IDs reserved for system functionality, are shown in the following table.

TABLE 4 User-configurable and reserved VLAN IDs for VDX devices

802.1Q VLANs	Classified VLANs (for Virtual Fabrics)
VLAN IDs 1 through 4086 (VLAN IDs 4087 through 4095 are reserved on these switches)	VLAN IDs 4096 through 8191 for service or transport VFs in a Virtual Fabrics context

## Debug and system diagnostic commands

Debug and system diagnostic commands, such as "debug" and "show system internal" commands, are developed and intended for specialized troubleshooting.

Extreme Networks recommends that you work closely with Extreme technical support in executing such commands and interpreting their results.

### NOTE

Not all diagnostic commands are documented.



# Commands A through E

---

## aaa accounting

Configures login or command accounting; either commands or login information are forwarded to accounting servers.

### Syntax

```
aaa accounting {commands defaultstart-stop [none | tacacs+] | execdefaultstart-stop [none | tacacs+]}
```

```
no aaa accounting {commands defaultstart-stop [none | tacacs+] | execdefaultstart-stop [none | tacacs+]}
```

### Parameters

#### commands

Toggles the logging of commands.

#### exec

Toggles the logging of login information.

#### default

Sends the logged information to the default server.

#### start-stop

Sends a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting server.

#### tacacs+

Sends the logged information to the TACACS+ server.

#### none

Disables accounting services.

### Modes

Global configuration mode

### Usage Guidelines

Use the **no aaa accounting** command to disable command accounting.

When logging commands, **show** commands are not forwarded.

### Examples

This example configures full accounting, with the CLI information being forwarded to the TACACS+ server.

```
device(config)# aaa accounting commands default start-stop tacacs+
```

This example disables login accounting, but leaves command accounting active.

```
device(config)# aaa accounting exec default start-stop none
```



# aaa authentication

Configures the AAA login sequence.

## Syntax

```
aaa authentication login { default | ldap | local }  
aaa authentication login { radius | tacacs+ } { local | local-auth-fallback }  
no aaa authentication login
```

## Command Default

The default server is Local.

## Parameters

### login

Specifies the type of server that will be used for authentication, authorization, and accounting (AAA) on the device. The local server is the default. Specify one of the following options:

#### default

Specifies the default mode (local server). Authenticates the user against the local database only. If the password does not match or the user is not defined, the login fails.

#### ldap

Specifies the Lightweight Directory Access Protocol (LDAP) servers.

#### local

Specifies to use the local device database if prior authentication methods are inactive.

#### radius

Specifies the RADIUS servers.

#### tacacs+

Specifies the TACACS+ servers.

#### local

Specifies to use the local device database if prior authentication methods are inactive.

#### local-auth-fallback

Specifies to use the local device database if prior authentication methods are not active or if authentication fails.

## Modes

Global configuration mode

## Usage Guidelines

This command selects the order of authentication sources to be used for user authentication during the login process. Two sources are supported: primary and secondary. The secondary source of authentication is optional and will be used if the primary source fails or is not available.

The authentication mode can only be set and cannot be added or deleted. For example, to change a configuration from "radius local" to radius only, execute the **no aaa authentication login** command to resets the configuration to the default mode, and then reconfigure the AAA mode with the desired setting.

In a configuration with primary and secondary sources of authentication, the primary mode cannot be modified alone. For example, you cannot change from "radius local" or "radius local-auth-fallback" to "tacacs+ local" or "tacacs+ local-auth-fallback" respectively. First remove the existing configuration and then configure it to the required configuration.

Beginning with Network OS v4.0.0, when the local option is specified as a secondary authentication service, local authentication is tried only when the primary AAA authentication service (TACACS+/Radius/LDAP) is either unreachable or not available. Local authentication will not be attempted if the authentication with the primary service fails.

## Examples

To change the AAA server to TACACS+ using the local device database as a secondary source of authentication:

```
device(config)# aaa authentication login tacacs+ local
Broadcast message from root (pts/0) Tue Apr  5 16:34:12 2011...
```

To change the AAA server from TACACS+ and local to TACACS+ only (no secondary source):

```
device(config)# no aaa authentication login tacacs+ local
device(config)# aaa authentication login tacacs+
device(config)# do show running-config aaa
aaa authentication login tacacs+
```

## Related Commands

[show running-config aaa](#)

# accept-unicast-arp-request

Configures the IPv4 fabric-virtual-gateway active sessions to respond to unicast ARP requests.

## Syntax

**accept-unicast-arp-request**

**no accept-unicast-arp-request**

## Modes

IPv4 address-family configuration mode

## Usage Guidelines

The **no accept-unicast-arp-request** command configures the active sessions to ignore ARP requests.

This command functions for IPv4 traffic only.

## Examples

The following example shows how to configure the gateway MAC address for an IPv4 Fabric-Virtual-Gateway session to respond to unicast ARP requests.

```
device(config)# router fabric-virtual-gateway
device(conf-router-fabric-virtual-gateway)# address-family ipv4
device(conf-address-family-ipv4)# accept-unicast-arp-request
```

## History

Release version	Command history
5.0.1	This command was introduced.

## access-group

Applies rules specified in an access control list (ACL) to traffic entering or exiting an interface. You can use the **no** form of this command to remove an ACL from an interface.

### Syntax

```
{ ip | ipv6 | mac } access-group ACLname { in | out } [ switched | routed ]
no { ip | ipv6 | mac } access-group ACLname { in | out } [ switched | routed ]
```

### Parameters

- ip**  
Specifies the IPv4 ACL to bind to an interface.
- ipv6**  
Specifies the IPv6 ACL to bind to an interface.
- mac**  
Specifies the Layer 2 ACL to bind to an interface.
- ACLname*  
Specifies the ACL name.
- in**  
Specifies the ACL binding direction as ingress.
- out**  
Specifies the ACL binding direction as egress.
- switched**  
Filter only switched traffic. This parameter is not valid for management or overlay-gateway interfaces.
- routed**  
Filter only routed traffic. This parameter is not valid for management or overlay-gateway interfaces.

### Modes

Interface subtype configuration mode

### Usage Guidelines

You can apply a maximum of six ACLs to a user interface, as follows:

- One ingress MAC ACL—if the interface is in switchport or overlay-gateway mode
- One egress MAC ACL—if the interface is in switchport mode
- One ingress IPv4 ACL
- One egress IPv4 ACL
- One ingress IPv6 ACL
- One egress IPv6 ACL

You can apply an ACL to multiple interfaces. And you can apply an extended ACL twice—ingress and egress—to a given user interface.

If you do not specify **switched** or **routed**, the ACL applies both to switched and routed traffic.

To remove an ACL from an interface, enter the **no** form of this command.

## Examples

```
device# configure terminal
Entering configuration mode terminal
device(config)# int ten 122/5/34
device(conf-if-te-122/5/34)# ip access-group ag1 in
```

# access-list

Creates a standard or extended access control list (ACL). In ACLs, you can define rules that permit or deny network traffic based on criteria that you specify.

## Syntax

```
{ ip| ipv6 | mac } access-list { standard | extended } ACLname
no { ip| ipv6 | mac } access-list { standard | extended } ACLname
```

## Parameters

ip	Specifies the Layer 2 or Layer 3 ACL type.
ipv6	Specifies the Layer 2 or Layer 3 ACL type.
mac	Specifies the Layer 2 or Layer 3 ACL type.
standard	Contains rules that permit or deny traffic based on source addresses that you specify. The rules are applicable to all ports of the specified addresses.
extended	Contains rules that permit or deny traffic according to source and destination addresses, as well as other parameters.
ACLname	Specifies a unique ACL name. The name can be up to 63 characters in length, and must begin with an alphanumeric character. No special characters are allowed, except for the underscore and hyphen.

## Modes

Global configuration mode

## Usage Guidelines

An ACL name can be up to 63 characters long, and must begin with a-z, A-Z or 0-9. You can also use underscore (\_) or hyphen (-) in an ACL name, but not as the first character.

On any given switch, an ACL name must be unique among all ACL types (MAC/IPv4/IPv6; standard or extended).

After you create an ACL, use the **seq** command to create filtering rules for that ACL.

An ACL starts functioning only after it is applied to an interface, using the **access-group** command.

To delete an ACL, use the **no access-list** command. You can delete an ACL only after you first remove it from all interfaces to which it is applied, using the **no access-group** command.

## Examples

The following example creates a standard IPv6 ACL:

```
device # configure terminal
device(config)# ipv6 access-list standard stdV6ACL1
```

The following example creates an extended IPv6 ACL:

```
device # configure terminal
device(config)# ipv6 access-list extended extdACL5
```

# action python-script

Specifies a Python file that runs when a trigger condition occurs.

## Syntax

**action python-script** *file-name*

**no action python-script** *file-name*

## Parameters

*file-name*

Specifies a Python script file-name. Valid values range from 4 through 32 characters (including the **.py** extension). The first character must be alphabetic.

## Modes

Event-handler configuration mode

## Usage Guidelines

You can assign only one action to a given event-handler profile.

You can also specify the Python file as part of the **event-handler** command.

To change the file assigned to a profile, you do not need to enter the **no** form of this command. You only need to enter **action python-script file-name**, specifying the new file name.

Running this command copies the Python script file from the `flash://` directory to the database. After specifying a file for all relevant event-handler profiles, you can delete it from the `flash://` directory.

If the event-handler for which you are modifying this command is active on one or more R Bridges, the changes take effect with no need to de-activate and re-activate the event-handler.

A Python event-handler script runs only if all of the following occur:

- Using the **copy** command, copy the Python file to the `flash://` location on the device.
- Using the **event-handler** command, create an event-handler profile.
- In configuration mode for that profile:
  - Using the **trigger** command, create one or more triggers.
  - Using the **action** command, specify the Python script that will be triggered.
- Using the **event-handler activate** command, activate an instance of the event handler.
- The trigger event occurs.

If an event-handler profile is not activated, the **no** form of this command deletes its action.



## Examples

The following example specifies Python files for two event-handler profiles.

```
device# configure terminal
device(config)# event-handler eventHandler1
device(config-event-handler-eventHandler1)# action python-script example.py
device(config-event-handler-eventHandler1)# event-handler eventHandler2
device(config-event-handler-eventHandler2)# action python-script example2.py
```

## History

Release version	Command history
6.0.1	This command was introduced.

# action-timeout

For an implementation of an event-handler profile, specifies the maximum number of minutes to wait for an action-script to complete execution. If the action-timeout expires, then script execution ends.

## Syntax

`action-timeout minutes`

`no action-timeout`

## Command Default

No action timeout is defined.

## Parameters

*minutes*

Specifies the number of minutes to wait for an action-script to complete execution. If you specify "0", no timeout is set. Valid timeout values are any positive integer.

## Modes

Event-handler activation mode

## Usage Guidelines

To restore the default setting of no timeout, enter the **no** form of this command.

## Examples

The following example specifies an action timeout of 30 minutes.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# action-timeout 30
```

## History

Release version	Command history
7.0.0	This command was introduced.

# activate (NSX Controller connection profile)

Activates an NSX controller connection profile, thereby initiating the connection between the NSX controller and the fabric.

## Syntax

**activate**  
**no activate**

## Command Default

Profile is inactive.

## Modes

NSX Controller configuration mode

## Usage Guidelines

You must configure the NSX Controller IP address before executing this command.

In a VCS Fabric, you must configure the VCS virtual IP address of the cluster before executing this command.

Use the **no** form of the command to mark the connection profile inactive. Any existing connection is closed. However, all tunnels already created by the NSX controller remain open.

## Examples

The following example activates an NSX controller connection profile that you have created and named profile1.

```
device# configure terminal
device(config)# nsx-controller profile1
device(config-nsx-controller-profile1)# activate
```

# activate (OpenFlow)

Activates an OpenFlow logical instance and configures OpenFlow behavior options.

## Syntax

```
activate [ controller name | default-behavior [ drop | send-to-controller ] | passive { no-ssl [ ip-address IPv4_address | port
  port-num ] } | version { ofv130 } ]
```

## Command Default

This feature is disabled.

## Parameters

### controller *name*

Specifies the name of an existing OpenFlow controller.

### default-behavior

Specifies the handling of table-miss packets.

#### drop

Drops packets in case of a table miss.

#### send-to-controller

Sends packets to the controller in case of a table miss.

### passive

Specifies the behavior for a passive connection.

#### no-ssl

Specifies no SSL connection.

#### ip-address *IPv4\_address*

Specifies an IPv4 address for the controller. See the Usage Guidelines.

#### port *port-num*

Specifies a TCP port to which remote controllers connect. Range is from 1 through 65535.

### version ofv130

Specifies the supported OpenFlow version (v1.3). This is the default and it does not need to be configured.

## Modes

OpenFlow logical-instance configuration mode

## Usage Guidelines

Only logical-instance 1 is supported.

If an IP address is not specified, any controller from any IP address can connect to the passive instance. With a specified IP address, only a controller with that address can connect.

The controller to activate must already be created by means of the **openflow-controller** command.

## Examples

The following example activates an OpenFlow logical instance.

```
device(config)# rbridge-id 1
device(config-rbridge-id-1)# openflow logical-instance 1
device(config-logical-instance-1)# activate
```

## History

Release version	Command history
6.0.1	This command was introduced.

# activate (OVSDB)

Activates an Open vSwitch Database SSL server for OpenStack deployments.

## Syntax

**activate**  
**no activate**

## Command Default

The server is not activated.

## Modes

OVSDB server configuration mode

## Usage Guidelines

Use the **no** form of this command to deactivate the SSL server.

## Examples

To activate an SSL server:

```
device# ovsdb-server myserver  
device(config-ovdsb-server-myserver)# activate
```

## History

Release version	Command history
7.0.0	This command was introduced.

# activate (protected VLAG)

Activates the port-channel redundancy group.

## Syntax

**activate**

**no activate**

## Modes

Global configuration mode

## Usage Guidelines

Use this command to activate the port-channel redundancy group to activate the protected VLAG. Once activated, no configuration changes are allowed on the protected VLAG and members.

The **no activate** command deactivates the protected VLAG.

## Examples

Typical command execution example:

```
device(config-port-channel-redundancy-group-32) # activate
```

## Related Commands

[port-channel](#), [port-channel-redundancy-group](#)

# activate (VXLAN gateway)

Activates a VXLAN overlay gateway instance.

## Syntax

**activate**  
**no activate**

## Command Default

By default, a gateway is not activated during initial configuration.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

It is recommended that you configure all gateway parameters before activating the gateway. This operation enables all tunnels that are associated with this gateway. VXLAN tunnels are not user configurable.

The following conditions that must be in place before you can execute the **activate** command:

- Loopback interfaces must be configured on all R Bridges that have been attached by means of the **attach** command. Refer to the **interface loopback** command.
- All loopback interfaces must be configured with the same IPv4 address and the same VRF instance.
- The IP address of the VXLAN gateway must be configured. Refer to the **ip interface** command.
- If attached R Bridges are configured for a VXLAN gateway, the VE, VRID and VRF configurations must match on all attached R Bridges.

Use the **no activate** command in VXLAN overlay gateway configuration mode to deactivate the gateway. All associated tunnels are also deactivated.

## Examples

The following example activates a VXLAN gateway named "gateway1". The gateway was previously configured by means of the **overlay-gateway** command:

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# activate
```

## Related Commands

[attach vlan](#), [interface loopback](#), [ip interface](#), [overlay-gateway](#)



# address-family

Enables IPv4 or IPv6 address-family configuration mode for Fabric-Virtual-Gateway at the VCS global level.

## Syntax

```
address-family { ipv4 | ipv6 }
no address-family { ipv4 | ipv6 }
```

## Command Default

None

## Modes

Fabric-Virtual-Gateway configuration mode

## Usage Guidelines

Enter the **no** form of the command to disable IPv4 or IPv6 address-family.

## Examples

The following example shows how to enable a Fabric-Virtual-Gateway session in Fabric-Virtual-Gateway address-family configuration mode.

```
device# configure terminal
device(config)# router fabric-virtual-gateway
device(conf-router-fabric-virtual-gateway)# address-family ipv4
device(conf-address-family-ipv4)# enable
```

## History

Release version	Command history
5.0.1	This command was introduced.

# address-family l2vpn evpn

Enables the L2VPN address family configuration mode to configure a variety of BGP EVPN options.

## Syntax

```
address-family l2vpn evpn
```

```
no address-family l2vpn evpn
```

## Modes

BGP configuration mode

## Usage Guidelines

Use this command in BGP configuration mode to enter BGP address-family L2VPN EVPN configuration mode. The L2VPN EVPN configuration mode supports the EVPN Subsequent Address Family Identifier (SAFI), an address qualifier that provides additional information about the Network Layer Reachability Information (NLRI) type for a given attribute. The **no** form of this command removes the L2VPN EVPN address family configuration from the device and removes all configurations under the L2VPN address family.

## Examples

The following example enables BGP address family L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)#
```

## History

Release version	Command history
7.0.0	This command was introduced.

# address-family unicast (BGP)

Enables the IPv4 or IPv6 address family configuration mode to configure a variety of BGP4 unicast routing options.

## Syntax

```
address-family { ipv4 | ipv6 } unicast [ vrf vrf-name ]
no address-family { ipv4 | ipv6 } unicast [ vrf vrf-name ]
```

## Parameters

- ipv4**  
Specifies an IPv4 address family.
- ipv6**  
Specifies an IPv6 address family.
- vrf vrf-name**  
Specifies the name of the VRF instance to associate with subsequent address-family configuration mode commands.

## Modes

BGP configuration mode

## Usage Guidelines

Use the **no** form of this command to remove IPv4 or IPv6 address family configurations from the device.

## Examples

This example enables BGP IPv4 address-family configuration mode:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)#
```

This example enables BGP IPv6 address-family configuration mode:

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)#
```

This example enables BGP IPv4 address-family configuration mode for VRF "green":

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf green
device(config-bgp-ipv4u-vrf)#
```

This example enables BGP IPv4 address-family configuration mode for VRF "red":

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)#
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	The command name was clarified with the addition of the required <b>unicast</b> keyword. The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# address-family unicast (VRF)

Enables the IPv4 or IPv6 address-family configuration mode to configure a variety of VRF unicast routing options.

## Syntax

```
address-family { ipv4 | ipv6 } unicast  
no address-family { ipv4 | ipv6 } unicast
```

## Modes

VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to remove IPv4 or IPv6 address-family configurations from the device.

## Examples

To enable IPv4 address-family configuration mode for VRF routing:

```
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# vrf orange  
device(config-vrf-orange)# address-family ipv4 unicast  
device(vrf-ipv4-unicast)#
```

To enable IPv6 address-family configuration mode for VRF routing:

```
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# vrf red  
device(config-vrf-red)# address-family ipv6 unicast  
device(vrf-ipv6-unicast)#
```

## Related Commands

[show vrf, vrf](#)

# advertise bgp-auto-nbr-tlv

Enables LLDP to advertise BGP information containing the interface IP address and Local AS number.

## Syntax

```
advertise bgp-auto-nbr-tlv
no advertise bgp-auto-nbr-tlv
```

## Command Default

Enabled.

## Modes

Interface subtype configuration mode  
 Protocol LLDP configuration mode  
 Profile configuration mode

## Usage Guidelines

The **no** form of the command disables advertising.

## Examples

The following example disables the advertising of BGP neighbor TLV in LLDP.

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)# no advertise bgp-auto-nbr-tlv
```

The following example disables the advertising of BGP neighbor TLV for a specific LLDP profile for a specified 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)# profile test1
device(conf-profile-test1)# no advertise bgp-auto-nbr-tlv
device(conf-profile-test1)# exit
device(conf-lldp)# exit
device(config)# interface fortygigabitethernet 1/0/49
device (conf-if-fo-1/0/49)# lldp profile test1
```

## History

Release version	Command history
7.2.0	This command was introduced.

# advertise dcbx-fcoe-app-tlv

Advertises application Type, Length, Values (TLVs) to ensure interoperability of traffic over the Data Center Bridging eXchange protocol (DCBX), which runs over LLDP to negotiate an FCoE application TLV.

## Syntax

```
advertise dcbx-fcoe-app-tlv
```

```
no advertise dcbx-fcoe-app-tlv
```

## Command Default

Advertisement is disabled.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

Converged Enhanced Ethernet (CEE) parameters related to FCoE must be negotiated before FCoE traffic can begin on a CEE link. An FCoE application TLV is exchanged over LLDP, which negotiates information such as FCoE priority, and Priority Flow Control (PFC) pause.

Enter **no advertise dcbx-fcoe-app-tlv** to return to the default setting.

## Related Commands

[advertise dot1-tlv](#), [advertise dot3-tlv](#), [advertise optional-tlv](#)

# advertise dcbx-fcoe-logical-link-tlv

Advertises to any attached device the FCoE status of the logical link.

## Syntax

`advertise dcbx-fcoe-logical-link-tlv`

`no advertise dcbx-fcoe-logical-link-tlv`

## Command Default

Advertisement is disabled.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

Enter `no advertise dcbx-fcoe-logical-link-tlv` to return to the default setting.

## Related Commands

[advertise dcbx-fcoe-app-tlv](#)



# advertise dcbx-iscsi-app-tlv

Advertises the iSCSI traffic configuration parameters for Type, Length, Values (TLV) values.

## Syntax

```
advertise dcbx-iscsi-app-tlv
```

```
no advertise dcbx-iscsi-app-tlv
```

## Command Default

Advertisement is enabled.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

No verification or enforcement of the usage of the advertised parameters by the iSCSI server or target is done by the switch.

Enter **no advertise dcbx-iscsi-app-tlv** to return to the default setting.

## Related Commands

[advertise dcbx-fcoe-app-tlv](#)

# advertise dcbx-tlv

Advertises to any attached device mandatory Data Center Bridging eXchange protocol (DCBX) Type, Length, Values (TLV) values.

## Syntax

`advertise dcbx-tlv`  
`no advertise dcbx-tlv`

## Command Default

Advertisement is enabled.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

Enter `no advertise dcbx-tlv` to return to the default setting.

# advertise dot1-tlv

Advertises globally to any attached device IEEE 802.1 organizationally specific Type, Length, Values (TLV) values, or for a specific LLDP profile.

## Syntax

```
advertise dot1-tlv
```

```
no advertise dot1-tlv
```

## Command Default

Advertisement is disabled.

## Modes

Protocol LLDP and profile configuration modes

## Usage Guidelines

Enter **no advertise dot1-tlv** to return to the default setting.

## Examples

The following example advertises TLV configuration for IEEE 802.1

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)# advertise dot1-tlv
device(conf-lldp)#
```

The following example advertises TLV configuration for IEEE 802.1 for a specific LLDP profile.

```
device(conf-lldp)# profile test1
device(config-profile-test1)# advertise dot1-tlv
device(conf-profile-test1)#
```

# advertise dot3-tlv

Advertises to any attached device IEEE 802.3 organizationally specific Type, Length, Values (TLV) values, or for a specific LLDP profile.

## Syntax

```
advertise dot3-tlv
no advertise dot3-tlv
```

## Command Default

Advertisement is disabled.

## Modes

Protocol LLDP and profile configuration modes.

## Usage Guidelines

Enter **no advertise dot3-tlv** to return to the default setting.

## Examples

The following example advertises TLV configuration for IEEE 802.3.

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)# advertise dot3-tlv
device(conf-lldp)#
```

The following example advertises TLV configuration for IEEE 802.3 for a specific LLDP profile.

```
device(conf-lldp)# profile test1
device(config-profile-test1)# advertise dot3-tlv
device(conf-profile-test1)#
```

## Related Commands

[advertise dcbx-tlv](#), [advertise dot1-tlv](#), [advertise optional-tlv](#)

# advertise optional-tlv

Advertises the optional Type, Length, and Values (TLV) values, or for a specific LLDP profile.

## Syntax

```
advertise optional-tlv { management-address | port-description | system-capabilities | system-description | system-name }  
no advertise optional-tlv
```

## Command Default

Advertisement is disabled.

## Parameters

### management-address

Advertises the management address of the system.

### port-description

Advertises the user-configured port.

### system-capabilities

Advertises the capabilities of the system.

### system-description

Advertises the system firmware version and the current image running on the system.

### system-name

Advertises the name of the system.

## Modes

Protocol LLDP and profile configuration modes

## Usage Guidelines

Enter **no advertise optional-tlv** to return to the default setting.

## Examples

The following example advertises the management address of the system and the user-configured port.

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)# advertise optional-tlv ?
Possible completions:
  management-address      Management Address TLV
  port-description        Port-Description TLV
  system-capabilities     System Capabilities TLV
  system-description      System Description
  system-name             System Name TLV
device(conf-lldp)# advertise optional-tlv management-address ?
Possible completions:
  port-description        Port-Description TLV
  system-capabilities     System Capabilities TLV
  system-description      System Description
  system-name             System Name TLV
device(conf-lldp)# advertise optional-tlv management-address port-description
device(conf-lldp)#
```

The following example advertises the management address of the system for a specific LLDP profile.

```
device(conf-lldp)# profile test1
device(config-profile-test1)# advertise optional-tlv ?
Possible completions:
  management-address      Management Address TLV
  port-description        Port-Description TLV
  system-capabilities     System Capabilities TLV
  system-description      System Description
  system-name             System Name TLV
device(conf-profile-test1)# advertise optional-tlv management-address
device(conf-profile-test1)#
```

## Related Commands

[advertise dcbx-tlv](#), [advertise dot1-tlv](#), [advertise dot3-tlv](#)

# advertise-backup

Enables a backup VRRP router to send advertisement frames to the master VRRP router.

## Syntax

```
advertise-backup
no advertise-backup
```

## Command Default

Advertisement is disabled.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

If a backup router is enabled to send advertisement frames, the frames are sent every 60 seconds.

This command can be used for VRRP-E, but not for VRRP.

Enter **no advertise backup** to return to the default setting (no periodic transmission).

## Examples

To enable the backup VRRP routers to send advertisement frames to the master VRRP router:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# interface ve 25
device(config-ve-25)# vrrp-extended-group 1
device(config-vrrp-extended-group-1)# advertise-backup
```

## Related Commands

[advertisement-interval \(VRRP\)](#), [backup-advertisement-interval](#), [vrrp-extended-group](#)

# advertisement interval (fabric-map)

Configures the FIP advertisement interval for the FCoE fabric-map mode.

## Syntax

**advertisement interval** *milliseconds*  
**no advertisement interval**

## Command Default

8000 milliseconds

## Parameters

*milliseconds*

The interval value in milliseconds. Valid values range from 250 through 90000 milliseconds.

## Modes

FCoE fabric-map configuration mode

## Usage Guidelines

You must be in the feature configuration mode for FCoE fabric-map for this command to function.

Enter **no advertisement interval** return to the default setting.

## Examples

```
device(config)# fcoe
device(config-fcoe)# fabric-map default
device(config-fcoe-fabric-map)# advertisement interval 8000
```

## Related Commands

[fcoe](#)



# advertisement-interval (VRRP)

Configures the interval at which the master VRRP router advertises its existence to the backup routers.

## Syntax

`advertisement-interval` *range*

## Command Default

1 second for version 2, 1000 milliseconds for version 3.

## Parameters

*range*

Interval at which the master VRRP router advertises its existence to the backup routers. Valid values range from 1 through 255 seconds for VRRPv2 and from 100 through 40900 milliseconds for VRRPv3.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

This interval is the length of time, in seconds, between each advertisement sent from the master to its backup VRRP routers. The advertisement notifies the backup routers that the master is still active. If the backup routers do not receive an advertisement from the master in a designated amount of time, the backup with the highest priority can assume the role of master.

This command can be used for either VRRP or VRRP-E and for VRRPv3 and VRRP-Ev3.

## Examples

To set the advertisement interval to 30 seconds for VRRP-E group 10:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# interface ve 25
device(config-ve-25)# vrrp-extended-group 10
device(config-vrrp-extended-group-10)# advertisement-interval 30
```

To set the advertisement interval to 3000 milliseconds for VRRP-Ev3 group 19:

```
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 2019
device(config-ve-2019)# ipv6 vrrp-extended-group 19
device(config-vrrp-extended-group-19)# advertisement-interval 3000
```

## Related Commands

[backup-advertisement-interval](#), [vrrp-extended-group](#), [vrrp-group](#)

# advertisement-interval-scale

Configures subsecond intervals at which the master VRRP-Ev3 device advertises its existence to the backup routers.

## Syntax

**advertisement-interval-scale** *scale*

## Command Default

The default advertisement interval scale is 1.

## Parameters

*scale*

Number representing the scale of the division of a configured interval at which the master VRRP-Ev3 device advertises its existence to the backup devices. Valid values are 1, 2, 5 and 10.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

This command scales the advertisement interval of the master VRRP-Ev3 device as configured by the **advertisement-interval** command. A value of 1, 2, 5, or 10 can be set and the existing advertisement interval value is divided by the scaling value, for example, if the advertisement interval is set to 1 second and the scaling value is set to 10, the new advertisement interval is 100 milliseconds. When all the advertisement intervals in a VRRP-Ev3 session are scaled, subsecond VRRP-Ev3 convergence is possible if a master fails. The advertisement notifies the backup devices that the master is still active. If the backup devices do not receive an advertisement from the master in a designated amount of time, the backup device with the highest priority can assume the role of master. Using subsecond advertising intervals, subsecond device redundancy can be achieved.

This command is only supported by VRRP-Ev3.

## Examples

To set the scaling of the advertisement interval to 500 milliseconds for VRRP-Ev3 group 19:

```
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 2019
device(config-ve-25)# ipv6 vrrp-extended-group 19
device(config-vrrp-extended-group-10)# advertisement-interval 1
device(config-vrrp-extended-group-10)# advertisement-interval-scale 2
```

## Related Commands

[advertisement-interval \(VRRP\)](#)

## ag

From RBridge ID configuration mode, accesses Access Gateway (AG) configuration mode, where you configure and enable Access Gateway.

## Syntax

ag

## Modes

RBridge ID configuration mode

## Usage Guidelines

From AG configuration mode, you can configure Access Gateway features such as Access Gateway policies, VF\_Port to N\_Port mapping, Port Grouping, N\_Port Monitoring reliability counters, and Modified Managed Fabric Name Monitoring (N-MFNM) mode timeout values. From AG configuration mode, you enable and disable Access Gateway.

## Examples

On RBridge 2, the following example changes to AG configuration mode.

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# ag
device(config-rbridge-id-2-ag)#
```

## History

Release version	Command history
6.0.1	<p>You can no longer append parameters to this command. But after you enter the <b>ag</b> command, switching to AG configuration mode, you can still enter these parameters as independent commands:</p> <ul style="list-style-type: none"> <li>• <b>enable</b></li> <li>• <b>counter reliability</b></li> <li>• <b>nport interface fiberchannel</b></li> <li>• <b>pg</b></li> <li>• <b>timeout fnm</b></li> </ul>

# aggregate-address (BGP)

Configures the device to aggregate routes from a range of networks into a single network prefix.

## Syntax

```
aggregate-address { ip-addr ip-mask | ipv6-addr ipv6-mask } [ advertise-map map-name ] [ as-set ] [ attribute-map map-name ] [ summary-only ] [ suppress-map map-name ]
```

```
no aggregate-address { ip-addr ip-mask | ipv6-addr ipv6-mask } [ advertise-map map-name ] [ as-set ] [ attribute-map map-name ] [ summary-only ] [ suppress-map map-name ]
```

## Command Default

The address aggregation feature is disabled. By default, the device advertises individual routes for all networks.

## Parameters

*ip-addr*

IPv4 address.

*ip-mask*

IPv4 mask.

*ipv6-addr*

IPv6 address.

*ipv6-mask*

IPv6 mask.

### **advertise-map**

Causes the device to advertise the more-specific routes in the specified route map.

*map-name*

Specifies a route map to be consulted.

### **as-set**

Causes the device to aggregate AS-path information for all routes in the aggregate routes from a range of networks into a single network prefix.

### **attribute-map**

Causes the device to set attributes for the aggregate routes according to the specified route map.

*map-name*

Specifies a route map to be consulted.

### **summary-only**

Prevents the device from advertising more-specific routes contained within the aggregate route.

### **suppress-map**

Prevents the more-specific routes contained in the specified route map from being advertised.

*map-name*

Specifies a route map to be consulted.

## Modes

BGP address-family IPv4 unicast configuration mode  
 BGP address-family IPv6 unicast configuration mode  
 BGP address-family IPv4 unicast VRF configuration mode  
 BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the defaults.

## Examples

This example aggregates routes from a range of networks into a single network prefix and prevents the device from advertising more-specific routes.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# aggregate-address 10.11.12.0 summary-only
```

This example aggregates routes from a range of networks into a single network prefix under the IPv6 address family and advertises the paths for this route as AS\_SET.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# aggregate-address 2001:DB8:12D:1300::/64 as-set
```

This example aggregates routes from a range of networks into a single network prefix for BGP VRF instance "red":

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# aggregate-address 5.0.0.0/8 .
```

This example aggregates routes from a range of networks into a single network prefix for BGP VRF instance "red" and prevents the device from advertising more-specific routes.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# aggregate-address 100.1.0.0/16 summary-only
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

## Related Commands

[route-map](#)

# alias

Configures global or user-level aliases for device commands.

## Syntax

**alias** *alias-name expansion*

**no alias** *alias-name*

## Parameters

*alias-name*

Specifies the alias name. The number of characters can be from 1 through 255.

*expansion*

Specifies the CLI command to be triggered when the alias is entered. If the command is more than one word, type double quotes (") around the command. The number of characters can be from 1 through 1023.

## Modes

Alias configuration mode

User-alias configuration mode

## Usage Guidelines

Global aliases are available to all users.

User-level aliases are available only for a specified user.

In the alias configuration mode, to delete a global alias use the **no** form of his command.

In the user-alias configuration mode, to delete a user alias use the **no** form of his command.

## Examples

The following example defines **ck** as a global alias that enters the **show clock** command.

```
device# configure terminal
device(config)# alias-config
device(config-alias-config)# alias ck "show clock"
```

For the user **jdoe**, the following example defines **sv** as a user-level alias that enters the **show version** command.

```
device# configure terminal
device(config)# alias-config
device(config-alias-config)# user jdoe
device(config-user-jdoe)# alias sv "show version"
```

## Related Commands

[alias-config, user \(alias configuration\)](#)

# alias-config

Launches the alias configuration mode, enabling you to define aliases.

## Syntax

```
alias-config
no alias-config [ alias | user username ]
```

## Parameters

**alias**  
(For the **no** option) Deletes all global aliases.

**user *username***  
(For the **no** option) Deletes all aliases defined for the specified user.

## Modes

Global configuration mode

## Usage Guidelines

From the alias configuration mode—which you access by entering this command—you can manage global aliases. From that mode, you can also access the user-alias configuration mode for a specified user, from which you can manage aliases for that user.

To delete all global aliases, use the **no alias-config alias** form of this command.

To delete all aliases defined for a specified user, use the **no alias-config user** form of this command.

## Examples

The following example accesses the alias configuration mode. It then defines `ck` as a global alias for the **show clock** command.

```
device# configure terminal
device(config)# alias-config
device(config-alias-config)# alias ck "show clock"
```

The following example deletes all aliases defined for the user `jdoe`.

```
device# configure terminal
device(config)# no alias-config user jdoe
```

## Related Commands

[alias](#), [user \(alias configuration\)](#)



# allow non-profiled-macs

Specifies whether non-profiled MAC addresses on the profiled port are dropped.

## Syntax

**allow non-profiled-macs**

**no allow non-profiled-macs**

## Command Default

Non-profiled MAC addresses are not dropped.

## Modes

Port-profile mode

## Usage Guidelines

This configuration is allowed on the default profile only.

Enter **no allow non-profiled-macs** to return to the default setting.

## Examples

```
device(config)# port-profile default
device(config-port-profile-default)# allow non-profiled-macs
```

# always-compare-med

Configures the device always to compare the Multi-Exit Discriminators (MEDs), regardless of the autonomous system (AS) information in the paths.

## Syntax

```
always-compare-med  
no always-compare-med
```

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command disallows the comparison of the MEDs for paths from neighbors in different autonomous systems.

## Examples

The following example configures the device always to compare the MEDs.

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# router bgp  
device(config-bgp-router)# always-compare-med
```

# always-propagate

Enables the device to reflect BGP routes even though they are not installed in the Routing Table Manager (RTM).

## Syntax

**always-propagate**

**no always-propagate**

## Command Default

This feature is disabled.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

## Examples

This example configures the device to reflect BGP routes that are not installed in the RTM.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# always-propagate
```

This example configures the device to reflect routes that are not installed in the RTM in a nondefault VRF instance.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# always-propagate
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# area authentication (OSPFv3)

Enables authentication for an OSPF Version 3 (OSPFv3) area.

## Syntax

```
area { A.B.C.D | decimal } authentication spi value { ah | esp null } { hmac-md5 | hmac-sha1 } key [ no-encrypt ] key
no area { A.B.C.D | decimal } authentication spi value
```

## Command Default

Authentication is not enabled on an area.

If the **no-encrypt** keyword is not used, the key is stored in encrypted format by default.

## Parameters

*A.B.C.D*

Area address in dotted decimal format.

*decimal*

Area address in decimal format.

**spi**

Specifies the Security Policy Index (SPI).

*value*

Specifies the Security Policy Index (SPI) value. Valid values range from decimal numbers 512 through 4294967295

**ah**

Specifies authentication header (ah) as the protocol to provide packet-level security.

**esp**

Specifies Encapsulating Security Payload (ESP) as the protocol to provide packet-level security.

**null**

Specifies that the ESP payload is not encrypted.

**hmac-md5**

Enables Hashed Message Authentication Code (HMAC) Message Digest 5 (MD5) authentication on the OSPF area.

**hmac-sha1**

Enables HMAC Secure Hash Algorithm 1 (SHA-1) authentication on the OSPF area.

**key**

Number used in the calculation of the message digest. The 40 hexadecimal character key is stored in encrypted format by default.

**no-encrypt**

The 40-character key is not encrypted upon either its entry or its display.

*key*

The 40 hexadecimal character key.

## Modes

- OSPFv3 router configuration mode
- OSPFv3 router VRF configuration mode

## Usage Guidelines

Enter **no area authentication spi** to remove an authentication specification for an area from the configuration.

The 40 hexadecimal character key is encrypted by default. Use the **no-encrypt** parameter to disable encryption.

## Examples

The following example enables ah and MD5 authentication for an OSPF area, setting a SPI value of 750.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip router-id 10.1.2.3
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# area 0 authentication spi 750 ah hmac-md5 key
abcef12345678901234fedcba098765432109876
```

The following example enables esp and SHA-1 authentication for an OSPF area, setting a SPI value of 900.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip router-id 10.1.2.3
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# area 0 authentication spi 900 esp null hmac-md5 sha1
abcef12345678901234fedcba098765432109876
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# area nssa (OSPFv2)

Creates a not-so-stubby area (NSSA) or modifies its parameters.

## Syntax

```
area { ip-addr | decimal } nssa metric [ no-summary ]
area { ip-addr | decimal } nssa default-information-metric metric num [ metric-type { type1 | type2 } ]
area { ip-addr | decimal } nssa default-information-metric metric-type { type1 | type2 } [ metric num ]
area { ip-addr | decimal } nssa no-redistribution
area { ip-addr | decimal } nssa translator-always
area { ip-addr | decimal } nssa translator-interval interval
no area nssa
```

## Command Default

No areas are created.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*metric*

Additional cost for using a route to or from this area. Valid values range from 1 through 6777215.

**no-summary**

When configured on the NSSA area border router (ABR), this parameter prevents any Type 3 and Type 4 summary link-state advertisement (LSA) from being injected into the area. The only exception is that a default route is injected into the NSSA by the ABR, and strictly as a Type 3 LSA (not a Type 7, because that could cause intra-AS traffic to get routed out the AS). This makes the NSSA an NSSA totally stubby area, which can only have Type 1, 2 and 7 LSAs.

**Note:** This parameter is disabled by default, which means the default route must use a Type 7 LSA.

**default-information-metric**

Specifies the OSPF default metric.

**metric num**

Specifies the OSPF route metric. Valid values range from 1 through 6777215.

**metric-type**

Metric type of default route originated into NSSA area.

**type1**

The metric of a neighbor is the cost between itself and the router plus the cost of using this router for routing to the rest of the world.

**type2**

The metric of a neighbor is the total cost from the redistributing routing to the rest of the world.

**default-information-originate**

When configured on the ABR, this parameter injects a Type 7 default route into the NSSA area. As a result, the other NSSA routers install the default route through the advertising NSSA ABR. By default the NSSA ABR does not originate a default route to the NSSA.

**no-redistribution**

The no-redistribution parameter prevents an NSSA ABR from generating external (type-7) LSA into a NSSA area. This is used in the case where an ASBR should generate type-5 LSA into normal areas and should not generate type-7 LSA into a NSSA area. By default, redistribution is enabled in a NSSA.

**translator-always**

Configures the translator role. When configured on an ABR, this causes the router to unconditionally assume the role of a NSSA translator. By default, translator-always is not set, the translator role by default is candidate.

**translator-interval** *interval*

Configures the time interval for which an elected NSSA translator continues to perform its duties even after its NSSA translator role has been disposed by another router. The range is 10 to 60 seconds and the default is 40.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

NSSAs are typically needed when one-way transmission of Type-5 LSAs (out of the area) is desired but injection of the same LSAs into the area is not acceptable.

Once created, the type of the area cannot be changed. The only exception to this rule is that an NSSA or stub area can be changed to a totally NSSA or a totally stub area, respectively.

The **no** form of the command deletes a NSSA.

## Examples

The following example sets an additional cost of 5 on an NSSA identified as 2 (in decimal format), and includes the no-summary parameter and prevents any Type 3 or Type 4 summary LSAs from being injected into the area.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 2 nssa 5 no-summary
```

The following specifies that the metric of a neighbor is the cost between itself and the router plus the cost of using this router for routing to the rest of the world.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 10.1.1.1 default-information-metric metric-type type1
```

The following example prevents an NSSA ABR from generating external LSAs into a NSSA area.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 10.1.1.1 nssa no-redistribution
```

The following example configures the router to unconditionally assume the role of a NSSA translator.

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 8 nssa translator-always
```

The following example configures a time interval of 20 seconds for which an elected NSSA translator continues to perform its duties even after its NSSA translator role has been disposed by another router.

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 10.1.1.1 nssa translator-interval 20
```

## History

Release version	Command history
7.0.1	This command was modified. The <b>default-information-metric</b> , <b>no-redistribution</b> , <b>translator-always</b> , and <b>translator-interval</b> parameters were added.



# area nssa (OSPFv3)

Creates a not-so-stubby area (NSSA) or modifies its parameters.

## Syntax

```
area { ip-addr | decimal } nssa [ metric ] [ default-information-originate [ metric num ] [ metric-type { type1 | type2 } ] ] [ no-
redistribution ] [ no-summary ] [ translator-always ] [ translator-interval interval ]
```

```
no area nssa
```

## Command Default

No areas are created.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*metric*

Additional cost for using a route to or from this area. Valid values range from 1 through 1048575.

### default-information-originate

When configured on the ABR, this parameter injects a Type 7 default route into the NSSA area. As a result, the other NSSA routers install the default route through the advertising NSSA ABR. By default the NSSA ABR does not originate a default route to the NSSA.

### metric-type

Specifies how the cost of a neighbor metric is determined.

#### type1

The metric of a neighbor is the cost between itself and the router plus the cost of using this router for routing to the rest of the world.

#### type2

The metric of a neighbor is the total cost from the redistributing routing to the rest of the world.

### no-redistribution

The no-redistribution parameter prevents an NSSA ABR from generating external (type-7) LSA into a NSSA area. This is used in the case where an ASBR should generate type-5 LSA into normal areas and should not generate type-7 LSA into a NSSA area. By default, redistribution is enabled in a NSSA.

### no-summary

When configured on the NSSA area border router (ABR), this parameter prevents any Type 3 and Type 4 summary link-state advertisement (LSA) from being injected into the area. The only exception is that a default route is injected into the NSSA by the ABR, and strictly as a Type 3 LSA (not a Type 7, because that could cause intra-AS traffic to get routed out the AS). This makes the NSSA a NSSA totally stubby area, which can only have Type 1, 2 and 7 LSAs.

**Note:** This parameter is disabled by default, which means the default route must use a Type 7 LSA.

**translator-always**

Configures the translator-role. When configured on an ABR, this causes the router to unconditionally assume the role of a NSSA translator. By default, translator-always is not set, the translator role by default is candidate.

**translator-interval** *interval*

Configures the time interval for which an elected NSSA translator continues to perform its duties even after its NSSA translator role has been disposed by another router. Valid values range from 10 through 60 seconds. By default the stability-interval is 40 seconds.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

NSSAs are typically needed when one-way transmission of Type-5 LSAs (out of the area) is desired but injection of the same LSAs into the area is not acceptable.

Once created, the type of the area cannot be changed. The only exception to this rule is that a NSSA or stub area can be changed to a totally NSSA or a totally stub area, respectively.

The **no** form of the command deletes a NSSA.

## Examples

The following example sets an additional cost of 4 on a NSSA identified as 8 (in decimal format), and prevents any Type 3 or Type 4 summary LSAs from being injected into the area.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# area 8 nssa 4 no-summary
```

## History

Release version	Command history
5.0.0	This command was introduced.

# area prefix-list (OSPFv2)

Filters prefixes advertised in type 3 link-state advertisements (LSAs) between OSPFv2 areas of an area border router (ABR).

## Syntax

**area** { *ip-addr* | *decimal* } **prefix-list** *name* { **in** | **out** }

**no area** { *ip-addr* | *decimal* } **prefix-list** *name* { **in** | **out** }

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

**prefix-list** *name*

Specifies a prefix-list between 1 and 32 characters.

**in**

Specifies that the prefix list is applied to prefixes advertised to the specified area from other areas.

**out**

Specifies that the prefix list is applied to prefixes advertised out of the specified area to other areas.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

This command is only applicable to ABRs. The **no** form of the command changes or cancels the configured filter and advertises all type 3 LSAs.

## Examples

The following example applies a prefix list to type 3 LSAs advertised out of an area with the area-id 10.1.1.1.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 10.1.1.1 prefix-list myprefixlist out
```

The following example applies a prefix list to type 3 LSAs advertised in to an area with the area-id 10.1.1.1.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 10.1.1.1 prefix-list myprefixlist in
```

## History

Release version	Command history
7.0.0	This command was introduced.

# area range (OSPFv2)

Specifies area range parameters on an area border router (ABR).

## Syntax

**area** { *A.B.C.D* | *decimal* } **range** *E.F.G.H I.J.K.L* **advertise** [ **cost** *cost\_value* ]

**area** { *A.B.C.D* | *decimal* } **range** *E.F.G.H I.J.K.L* **not-advertise** [ **cost** *cost\_value* ]

**area** { *A.B.C.D* | *decimal* } **range** *E.F.G.H I.J.K.L* **cost** *cost\_value*

**no area range**

## Parameters

*A.B.C.D*

Area address in IP address format.

*decimal*

Area address in decimal format.

*E.F.G.H I.J.K.L*

Specifies the IP address and mask portion of the range. All network addresses that match this network are summarized in a single route and advertised by the ABR.

**advertise**

Sets the address range status to *advertise* and generates a Type 3 summary LSA.

**cost** *cost\_value*

Sets the cost value for the area range. This value is used as the generated summary LSA cost. The range for *cost\_value* is 1 to 6777214. If this value is not specified, the cost value is the default range metric calculation for the generated summary LSA cost.

**not-advertise**

Sets the address range status to DoNotAdvertise; the Type 3 LSA is suppressed, and the component networks remain hidden from other networks. This setting is used to temporarily pause route summarization from the area.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use this command only on ABRs to specify route summarization for an existing area. The result is that a single summary route is advertised to other areas by the ABR, in the form of a Type 3 LSA. Routing information is condensed at area boundaries and external to the area, and only a single route is advertised for each address range.

An example of when you might want to use this command is if you have many small networks advertised from area 0 to any other area, or from any non-backbone area into the backbone. This command gives you a summary route instead of many

smaller routes. In an area, the OSPF database on each router must be an exact copy of the databases of the other routers. This means that no summarization is allowed within the area.

The **no** form of the command disables the specification of range parameters on an ABR.

## Examples

The following example advertises to Area 3 all the addresses on the network 10.1.1.0 10.255.255.0 in the ABR you are signed into.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 3 range 10.1.1.0 10.255.255.0 advertise
```

## History

Release version	Command history
5.0.0	Support was added for OSPFv3.

# area stub (OSPFv2)

Creates or deletes a stub area or modifies its parameters.

## Syntax

```
area { ip-addr | decimal } stub metric [ no-summary ]  
no area stub
```

## Command Default

No areas are created.

## Parameters

*A.B.C.D*

Area address in IP address format.

*decimal*

Area address in decimal format.

*metric*

Additional cost for using a route to or from this area. Valid values range from 1 through 6777215.

**no-summary**

When configured on the ABR, this parameter prevents any Type 3 and Type 4 summary LSAs from being injected into the area. The only exception is that a default route is injected into the stub/totally stubby area by the ABR as a Type 3 LSA. Enabling this parameter makes the area a so-called totally stubby area, which can only have Types 1 and 2. This parameter is disabled by default.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Once created, the type of the area cannot be changed. The only exception to this rule is that a NSSA or stub area can be changed to a totally NSSA or a totally stub area, respectively.

The **no** form of the command deletes a stub area.

## Examples

The following example sets an additional cost of 5 on a stub area called 2.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 2 stub 5
```



# area virtual-link (OSPFv2)

Creates or modifies virtual links for an area.

## Syntax

```
area { ip-addr | decimal } virtual-link E.F.G.H [ authentication-key { 0 | 2 | 255 } password ] [ dead-interval time ] [ hello-interval time ] [ md5-authentication { key-activation-wait-time time | key-id num key } ] [ retransmit-interval time ] [ transmit-delay time ]
```

```
no area virtual-link
```

## Command Default

No virtual links are created.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*E.F.G.H*

ID of the OSPF router at the remote end of the virtual link.

### authentication-key

Sets the password and encryption method. Only one encryption method can be active on an interface at a time. All OSPF packets transmitted on the interface contain this password. All OSPF packets received on the interface are checked for this password. If the password is not present, then the packet is dropped.

**0**

Does not encrypt the password you enter.

**2**

Encrypts the password you enter.

**255**

Encrypts a plain-text password that you enter.

*password*

OSPF password. The password can be up to eight alphanumeric characters.

### dead-interval *time*

How long a neighbor router waits for a hello packet from the current router before declaring the router down. This value must be the same for all routers and access servers that are attached to a common network. Valid values range from 3 through 65535 seconds. The default is 40 seconds.

### hello-interval *time*

Time between hello packets that the router sends on an interface. The value must be the same for all routers and access servers that are attached to a common network. Valid values range from 1 through 65535 seconds. The default is 10 seconds.

**md5-authentication**

Sets either MD5 key-activation wait time or key identifier.

**key-activation-wait-time** *time*

Time before a newly configured MD5 authentication key is valid. This parameter provides a graceful transition from one MD5 key to another without disturbing the network. All new packets transmitted after the wait time ends will use the newly configured MD5 Key. OSPF packets that contain the old MD5 key are accepted for up to five minutes (300 seconds) after the new MD5 key is in operation. Valid values range from 0 through 14400 seconds. The default is 300 seconds.

**key-id** *num key*

The *num* is a number between 1 and 255 which identifies the MD5 key being used. This parameter is required to differentiate among multiple keys defined on a device. When MD5 is enabled, the key is an alphanumeric password of up to 16 characters that is later encrypted and included in each OSPF packet transmitted. You must enter a password in this field when the system is configured to operate with either simple or MD5 authentication. By default, the MD5 authentication key is encrypted.

**retransmit-interval** *time*

Time between Link State Advertisement (LSA) retransmissions for adjacencies belonging to the interface. Set this interval to a value larger than the expected round-trip delay between any two routers on the attached network. Valid values range from 0 through 3600 seconds. The default is 5 seconds.

**transmit-delay** *time*

Estimated time required to send an LSA on the interface. This value must be an integer greater than zero. The age of each LSA in the update packet is incremented by the value of this parameter before transmission occurs. Valid values range from 0 through 3600 seconds. The default is 1 second.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

The **no** form of the command removes a virtual link.

## Examples

The following example creates a virtual link for an area whose decimal address is 1, and where the ID of the OSPFv2 device at the remote end of the virtual link is 10.1.2.3.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# area 1 virtual-link 10.1.2.3
```

# area virtual-link (OSPFv3)

Creates or modifies virtual links for an area.

## Syntax

```
area { ip-addr | decimal } virtual-link A.B.C.D [ dead-interval time | hello-interval time | hello-jitter interval | retransmit-interval time | transmit-delay time ]
```

```
no area virtual-link
```

## Command Default

No virtual links are created.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*A.B.C.D*

ID of the OSPFv3 device at the remote end of the virtual link.

**dead-interval** *time*

How long a neighbor device waits for a hello packet from the current device before declaring the device down. This value must be the same for all devices and access servers that are attached to a common network. Valid values range from 3 through 65535 seconds. The default is 40 seconds.

**hello-interval** *time*

Time between hello packets that the device sends on an interface. The value must be the same for all devices and access servers that are attached to a common network. Valid values range from 1 through 65535 seconds. The default is 10 seconds.

**hello-jitter** *interval*

Sets the allowed jitter between hello packets. Valid values range from 1 through 50 percent (%). The default value is 10%.

**retransmit-interval** *time*

Time between Link State Advertisement (LSA) retransmissions for adjacencies belonging to the interface. Set this interval to a value larger than the expected round-trip delay between any two devices on the attached network. Valid values range from 0 through 3600 seconds. The default is 5 seconds.

**transmit-delay** *time*

Estimated time required to send an LSA on the interface. This value must be an integer greater than zero. The age of each LSA in the update packet is incremented by the value of this parameter before transmission occurs. Valid values range from 0 through 3600 seconds. The default is 1 second.

## Modes

- OSPFv3 router configuration mode
- OSPFv3 router VRF configuration mode

## Usage Guidelines

The values of the **dead-interval** and **hello-interval** parameters must be the same at both ends of a virtual link. Therefore, if you modify the values of these parameters at one end of a virtual link, you must make the same modifications on the other end of the link. The values of the other virtual link parameters do not require synchronization.

The **no** form of the command removes a virtual link.

## Examples

The following example creates a virtual link for an area whose decimal address is 1, and where the ID of the OSPFv3 device at the remote end of the virtual link is 209.157.22.1.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# area 1 virtual-link 209.157.22.1
```

## History

Release version	Command history
5.0.0	This command was introduced.

# area virtual-link authentication (OSPFv3)

Enables authentication for virtual links in an OSPFv3 area.

## Syntax

```
area { ip-addr | decimal } virtual-link E.F.G.H authentication spi spi { ah | esp null } { hmac-md5 | hmac-sha1 } key [ no-encrypt ] key
```

```
no area { A.B.C.D | decimal } virtual-link E.F.G.H authentication spi spi
```

## Command Default

Authentication is not enabled on a virtual-link.

The 40 hexadecimal character key is encrypted by default. Use the **no-encrypt** parameter to disable encryption.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*E.F.G.H*

ID of the OSPFv3 device at the remote end of the virtual link.

**spi**

Specifies the security policy index (SPI) value. Valid values range from decimal numbers 512 through 4294967295

**ah**

Specifies authentication header (ah) as the protocol to provide packet-level security.

**esp**

Specifies Encapsulating Security Payload (ESP) as the protocol to provide packet-level security.

**null**

Specifies that the ESP payload is not encrypted.

**hmac-md5**

Enables Hashed Message Authentication Code (HMAC) Message Digest 5 (MD5) authentication on the OSPF area.

**hmac-sha1**

Enables HMAC Secure Hash Algorithm 1 (SHA-1) authentication on the OSPF area.

**key**

Number used in the calculation of the message digest. The 40 hexadecimal character key is stored in encrypted format by default.

**no-encrypt**

The 40-character key is not encrypted upon either its entry or its display.

*key*

The 40 hexadecimal character key.

## Modes

- OSPFv3 router configuration mode
- OSPFv3 router VRF configuration mode

## Usage Guidelines

Enter **no area** { *A.B.C.D* | *decimal* } **virtual-link** *E.F.G.H* **authentication spi spi** to remove authentication from the virtual-links in the area.

## Examples

This example configures IPsec on a virtual link in an OSPFv3 area.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip router-id 10.1.2.2
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# area 2 virtual-link 10.1.2.2 authentication spi 600 ah
  hmac-sha1 no-encrypt key 1134567890223456789012345678901234567890
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# arp

Creates a static Address Resolution Protocol (ARP) entry.

## Syntax

```
arp A.B.C.D mac-address interface { <N>gigabitethernet rbridge-id / slot / port | port-channel number | ve vlan-id }
no arp A.B.C.D
```

## Parameters

*A.B.C.D*

Specifies a valid IP address.

*mac-address*

Specifies a valid MAC address.

**interface**

Specifies an interface type.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel. Range is from 1 through 6144. The port-channel must be configured with an IP address.

**ve** *vlan-id*

Specifies the corresponding VLAN interface that must already be configured before the VE interface can be created. Refer to the Usage Guidelines.

## Modes

RBridge ID configuration mode

VRF configuration mode

## Usage Guidelines

The **no** form of the command deletes a static ARP entry.

Before you can configure a VE interface, you must configure a VLAN interface. The corresponding VE interface must use the same VLAN ID you used to configure the VLAN.

To remove the VE interface, enter **no interface ve *vlan-id***. Note that this does not remove the corresponding VLAN interface.



#### CAUTION

If no RBridge ID is configured on the switch, deleting the VE interface will cause a spike in CPU usage. To prevent this, configure an RBridge ID before deleting the VE interface.

## Examples

The following example configures the 2/0/33 interface as a member of port-channel 20 and then defines a static ARP for that port channel.

```
device# configure terminal
device(config)# interface tengigabitethernet 2/0/33
device(conf-if-te-2/0/33)# channel-group 20 mode on
device(conf-if-te-2/0/33)# exit
device(config)# rbridge-id 2
device(config-rbridge-id-2)# arp 2.2.2.2 1232.1211.1333 interface Port-channel 20
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.



# arp access-list

Creates an address resolution protocol (ARP) access list (ACL), which is one of the steps implementing dynamic ARP inspection (DAI) on a VLAN.

## Syntax

```
arp access-list acl-name
```

```
no arp access-list acl-name
```

## Command Default

No ARP access lists are defined.

## Parameters

*acl-name*

Specifies the name of the ARP ACL. The name can be up to 63 characters in length, and must begin with an alphanumeric character. No special characters are allowed, except for the underscore and hyphen.

## Modes

Global configuration mode

## Usage Guidelines

You can also append the **permit ip host** command to this command.

If the ACL is not applied on any VLAN, the **no form** of this command deletes the ACL.

On untrusted interfaces of DAI-enabled VLANs, incoming ARP packets from permitted IP/MAC addresses are accepted only if all of the following steps were performed:

- Create the ACL, using the **arp access-list** command.
- In the ACL, create one or more rules, using the **permit ip host** command. Each rule specifies an IP/MAC address-pair.
- Apply the ACL to one or more VLANs, using the **ip arp inspection filter** command.
- Enable DAI on such VLANs, using the **ip arp inspection** command.

## Examples

The following example creates an ARP ACL named "host2" and then defines one **permit** rule in that ACL.

```
device# configure terminal
device(config)# arp access-list host2
device(config-arp-acl)# permit ip host 1.1.1.1 mac host 0000.0011.0022
```

## History

Release version	Command history
6.0.1	This command was introduced.

# as-path-ignore

Disables the comparison of the autonomous system (AS) path lengths of otherwise equal paths.

## Syntax

`as-path-ignore`

`no as-path-ignore`

## Command Default

The comparison of the AS path lengths of otherwise equal paths is enabled.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command restores default behavior.

## Examples

The following example configures the device to always disable the comparison of AS path lengths.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# as-path-ignore
```

# attach rbridge-id

Assigns a range of RBridge IDs to a VXLAN gateway instance.

## Syntax

```
attach rbridge-id { add | remove } rb-range
```

## Parameters

### add

Attaches a specified range of RBridge IDs to a VXLAN gateway.

### remove

Un-attaches a specified range of RBridge IDs from a VXLAN gateway.

### *rb-range*

Specifies a range of RBridge IDs to attach to the VXLAN gateway, up to a maximum of four RBridge IDs. (You can also specify just one RBridge ID.) Ranges can be specified by hyphens, separated by commas, or contain a mixture of both.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

Use the **add** form of the command to attach RBridge IDs to the VXLAN gateway, and use the **remove** form of the command to unattach RBridge IDs from the VXLAN gateway. The maximum number of RBridge IDs that can be attached is four.

When unattaching RBridge IDs, gateway and tunnel configurations on the specified RBridge IDs are deleted.

You can configure other properties for the gateway instance while in VXLAN overlay gateway configuration mode, but the gateway instance is not created until you enter the **attach rbridge-id** command. Do not use a space after a comma when specifying a range of RBridge IDs. For example, to specify RBridges 5 through 7 and RBridge 9, enter the following: 5-7,9.

The RBridge IDs that you specify must already be known to the cluster. (RBridge IDs that have been removed from the cluster by means of the **no vcs enable rbridge-id** command cannot be used to attach to the VXLAN gateway.)

The RBridge IDs that you specify must be on a VXLAN-capable gateway (either the VDX 6740, VDX 6740-T and VDX 6940).

## Examples

The following example adds an RBridge ID range of 10 through 12 on a VXLAN overlay gateway instance named "gateway1".

```
device# configure
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# attach rbridge-id add 10-12
```

## Related Commands

[overlay-gateway](#), [rbridge-id](#)

# attach rbridge-id (Fabric-Virtual-Gateway)

Assigns a range of RBridge IDs to the global VE interface.

## Syntax

```
attach rbridge-id { add | remove } rb-range
```

## Command Default

None

## Parameters

### add

Attaches a specified range of RBridge IDs to the VE interface.

### remove

Removes a specified range of RBridge IDs from the VE interface.

### *rb-range*

Specifies a range of RBridge IDs to attach to the VE interface, up to a maximum of four RBridge IDs. (You can also specify a single RBridge ID.) Ranges can be specified by hyphens, separated by commas, or contain a mixture of both.

## Modes

Fabric-Virtual-Gateway global VE interface configuration mode

## Usage Guidelines

Use the **add** form of the command to attach RBridge IDs to the VE interface, and use the **remove** form of the command to unattach RBridge IDs from the VE interface. The maximum number of RBridge IDs that can be attached is four.

## Examples

The following example shows how to attach an RBridge-ID to the VE interface.

```
device# configure terminal
device(config)# interface ve 2000
device(config-Ve-2000)# attach rbridge-id add 54,55
```

## History

Release version	Command history
5.0.1	This command was introduced.

# attach vlan

Identifies exported VLANs in VXLAN gateway configurations.

## Syntax

```
attach vlan vlan_ID [ mac mac_address ]
no attach vlan vlan_ID
```

## Parameters

**vlan** *vlan\_ID*  
Specifies the VLAN ID of the VXLAN gateway. This can be a range, such as 5, 10, 20-25.

**mac** *mac\_address*  
Specifies the MAC address of a VXLAN gateway, in *HHHH.HHHH.HHHH* format.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

Exported VLANs are VLANs that can be mapped to VXLAN domains. All the MAC addresses that the VXLAN gateway learns on these VLANs are shared with the NSX controller.

This command is applicable only when the gateway type is **nsx**, as configured by means of the **overlay-gateway** command.

This command can optionally accept specific MAC addresses, which can be shared with the NSX Controller. If the user specifies MAC addresses, only the specified MAC addresses are shared with the NSX Controller for the specified VLAN. The specified VLAN must already be configured.

You cannot run two forms of this command that use the same VLAN IDs. For example, the commands **attach vlan x** and **attach vlan xmac y** cannot coexist. If one form of the configuration exists, the other form of the configuration that uses the same VLAN ID is rejected.

Also, you cannot specify a VLAN range and a MAC address on the same command line. You can, however, specify a single VLAN ID and a MAC address on the same command line.

The **no** form of this command stops the MAC addresses behind the specified VLANs from being shared with the NSX Controller.

The deletion of a VLAN specified by this command is not allowed. For example, if you enter the **attach vlan x** command, you cannot delete the exported VLAN called x by running the **no interface vlan x** command.

## Examples

To specify an exported VLAN ID and a MAC address for a VXLAN gateway named "gateway1" that is already configured:

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# attach vlan 5 mac 00:05:1e:c5:96:a4
```

attach vlan

## Related Commands

[overlay-gateway](#)



# auto-config-backup

Initiates a manual backup, from global configuration mode, of a running-configuration master file across all nodes in the cluster.

## Syntax

```
auto-config-backup
```

## Command Default

This feature is disabled.

## Modes

Global configuration mode

## Usage Guidelines

You can also use the **vcs auto-config-backup timer** command in global configuration mode to initiate a running-configuration backup across all nodes in the cluster.

### ATTENTION

Whichever command initiates the most recent backup produces this file, overwriting the single master backup file that is maintained internally.

## Examples

To initiate from global configuration mode a manual backup of the running-configuration master file across all nodes in the cluster:

```
device# configure terminal
device(config)# auto-config-backup
```

## History

Release version	Command history
7.1.0	This command was introduced.

## Related Commands

[vcs auto-config-backup timer](#), [show vcs auto-config-backup](#)

# auto-cost reference-bandwidth (OSPFv2)

Configures reference bandwidth.

## Syntax

```
auto-cost reference-bandwidth { value | use-active-ports }  
no auto-cost reference-bandwidth
```

## Command Default

Reference bandwidth is 100 Mbps.

## Parameters

*value*

Reference bandwidth in Mbps. Valid values range from 1 through 4294967.

**use-active-ports**

Specifies that any dynamic change in bandwidth immediately affects the cost of OSPF routes. This parameter enables cost calculation for currently active ports only.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use this command to configure the cost of an interface that a device advertises to its OSPF neighbors. OSPF calculates the cost of a route as the ratio of the reference bandwidth to the bandwidth of the egress interface. An increase in the reference bandwidth results in an increased cost. If the resulting cost is less than 1, the software rounds the cost up to 1.

The bandwidth for interfaces that consist of more than one physical port is calculated as follows:

- LAG group — The combined bandwidth of all the ports.
- Virtual interface — The lowest individual bandwidth of all the ports that carry the VLAN for the associated VE.

If a change to the reference bandwidth results in a cost change to an interface, the device sends a link-state update to update the costs of interfaces advertised by the device.

### NOTE

If you specify the cost for an individual interface (by using the **ip ospf cost** command), the cost you specify overrides the cost calculated by the software.

The **no** form of the command disables bandwidth configuration.

## Examples

The following example configures a reference bandwidth of 500.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# auto-cost reference-bandwidth 500
```

The reference bandwidth specified in this example results in the following costs:

- 10 Mbps port's cost =  $500/10 = 50$ .
- 100 Mbps port's cost =  $500/100 = 5$ .
- 1000 Mbps port's cost =  $500/1000 = 0.5$ , which is rounded up to 1.

The costs for 10 Mbps and 100 Mbps ports change as a result of the changed reference bandwidth. Costs for higher-speed interfaces remain the same.

## Related Commands

[ip ospf cost](#)

# auto-cost reference-bandwidth (OSPFv3)

Configures reference bandwidth.

## Syntax

```
auto-cost reference-bandwidth value  
no auto-cost reference-bandwidth
```

## Command Default

Reference bandwidth is 100 Mbps.

## Parameters

*value*

Reference bandwidth in Mbps. Valid values range from 1 through 4294967. The default is 100 Mbps.

## Modes

OSPFv3 router configuration mode  
OSPFv3 router VRF configuration mode

## Usage Guidelines

Use this command to configure the cost of an interface that a device advertises to its OSPF neighbors. OSPFv3 calculates the cost of a route as the ratio of the reference bandwidth to the bandwidth of the egress interface. An increase in the reference bandwidth results in an increased cost. If the resulting cost is less than 1, the software rounds the cost up to 1.

The bandwidth for interfaces that consist of more than one physical port is calculated as follows:

- LAG group — The combined bandwidth of all the ports.
- Virtual interface — The lowest individual bandwidth of all the ports that carry the VLAN for the associated VE.

If a change to the reference bandwidth results in a cost change to an interface, the device sends a link-state update to update the costs of interfaces advertised by the device.

### NOTE

If you specify the cost for an individual interface using the **ipv6 ospf cost** command, the cost you specify overrides the cost calculated by the software.

The **no** form of the command restores the reference bandwidth to its default value and, thus, restores the default costs of the interfaces to their default values.

## Examples

The following example configures a reference bandwidth of 500.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# auto-cost reference-bandwidth 500
```

The reference bandwidth specified in this example results in the following costs:

- 10 Mbps port's cost =  $500/10 = 50$ .
- 100 Mbps port's cost =  $500/100 = 5$ .
- 1000 Mbps port's cost =  $500/1000 = 0.5$ , which is rounded up to 1.
- 155 Mbps port cost =  $500/155 = 3.23$ , which is rounded up to 4
- 622 Mbps port cost =  $500/622 = 0.80$ , which is rounded up to 1
- 2488 Mbps port cost =  $500/2488 = 0.20$ , which is rounded up to 1

The costs for 10 Mbps, 100 Mbps, and 155 Mbps ports change as a result of the changed reference bandwidth. Costs for higher-speed interfaces remain the same.

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[ipv6 ospf cost](#)

# auto-shutdown-new-neighbors

Disables the establishment of BGP connections with a remote peer when the peer is first configured.

## Syntax

**auto-shutdown-new-neighbors**

**no auto-shutdown-new-neighbors**

## Command Default

This feature is disabled.

## Modes

BGP configuration mode

## Usage Guidelines

The **auto-shutdown-new-neighbors** command applies to all neighbors configured under each VRF. When the **auto-shutdown-new-neighbors** command is used, any new neighbor configured will have the shutdown flag enabled for them by default. Once all the neighbor parameters are configured and it is ready to start the establishment of BGP session with the remote peer, the BGP neighbor's shutdown parameter has to disabled by removing the shutdown command for the neighbor.

The **no** form of the command restores the default.

## Examples

The following example enables auto shutdown of BGP neighbors on initial configuration.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# auto-shutdown-new-neighbors
```

The following example disables the peer shutdown state and begins the BGP4 session establishment process.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# local-as 65520
device(config-bgp-router)# no neighbor 10.1.1.1 shutdown
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[neighbor shutdown](#)

# backup-advertisement-interval

Configures the interval at which backup VRRP routers advertise their existence to the master router.

## Syntax

```
backup-advertisement-interval interval
```

## Command Default

The default backup advertisement-interval is 60 seconds.

## Parameters

*interval*

Interval at which a backup VRRP router advertises its existence to the master router. Valid values range from 60 through 3600 seconds.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

The interval is the length of time, in seconds, between each advertisement sent from the backup routers to the master router. The advertisement notifies the master router that the backup is still active. If the master router does not receive an advertisement from the backup in a designated amount of time, the backup with the highest priority can assume the role of master.

This command can be used for either VRRP or VRRP-E.

## Examples

To set the backup advertisement interval to 120 seconds for VRRP-E group 10:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# interface ve 25
device(config-ve-25)# vrrp-extended-group 10
device(config-vrrp-extended-group-10)# backup-advertisement-interval 120
```

## Related Commands

[vrrp-extended-group](#), [vrrp-group](#)



# banner incoming

Sets the incoming banner message.

## Syntax

**banner incoming** *message*

**no banner incoming**

## Parameters

*message*

The message string to be displayed on the switch console.

## Modes

Global configuration mode

## Usage Guidelines

A banner is a text message that displays on the console. The banner can include information about the switch for a user to know when accessing the switch.

The banner must be from 1 through 2048 characters in length. The banner can appear on multiple lines if you enter multiline mode using **ESC+M** and using **CTRL+D** to exit.

# banner login

Sets the switch banner.

## Syntax

**banner login** *message*

**no banner login**

## Parameters

*message*

The message string to be displayed on the switch console.

## Modes

Global configuration mode

## Usage Guidelines

A banner is a text message that displays on the console. The banner can include information about the switch that a user wants another user to know when accessing the switch.

The banner must be from 1 through 2048 characters in length.

The banner can appear on multiple lines if you enter multiline mode using **ESC-M** and using **CTRL-D** to exit.

## Examples

To create a banner with multiple lines:

```
device# configure terminal
device(config)# banner login [Esc-m]

[Entering multiline mode, exit with ctrl-D.]
> banner login Hello
> and
> welcome
> to
> the
> switch
[Ctrl-D]

device(config)# do show running-config banner

banner login "Hello\and\welcome\to\the\switch"

device(config)# exit

Network OS (switch)
NOS Version 3.0.0
switch login: admin

Password: *****

Hello and welcome to the switch
```

To create a banner with a single line:

```
device# configure terminal
device(config)# banner login "Please do not disturb the setup on this switch"

device(config)# exit

Login: user

Password: *****

The cluster contains 5 switches
-----
Welcome to NOS CLI
user connected from ::FFFF:10.103.8.61 using ssh on abc.com
device#
```

## Related Commands

[show running-config banner](#)

# banner motd

Sets the message of the day (MOTD) banner.

## Syntax

`banner motd message`

`no banner motd`

## Parameters

*message*

The message string to be displayed on the switch console.

## Modes

Global configuration mode

## Usage Guidelines

A banner is a text message that displays on the console. The banner can include information about the switch for a user to know when accessing the switch.

The banner must be from 1 through 2048 characters in length. The banner can appear on multiple lines if you enter multiline mode by using **ESC+M** and exit by using **CTRL+D** .

# beacon

Enables the flashing LED beacon on the switch, which makes it easier to find the specified switch chassis or Ethernet interface port in large data centers.

## Syntax

```
beacon { enable | disable [ chassis | interface { fibrechannel rbridge-id/slot/port | fortygigabitethernet rbridge-id/slot/port |
gigabitethernet rbridge-id/slot/port | hundredgigabitethernet rbridge-id/slot/port | tengigabitethernet rbridge-id/slot/
port } disable [ chassis | interface { fortygigabitethernet rbridge-id/slot/port | gigabitethernet rbridge-id/slot/port |
hundredgigabitethernet rbridge-id/slot/port | tengigabitethernet rbridge-id/slot/port } }
```

## Command Default

The LED is disabled.

## Parameters

### enable

Enables the beacon for an entire chassis or an Ethernet interface (and corresponding port).

### disable

Enables the beacon for an entire chassis or an Ethernet interface (and corresponding port).

### chassis

Specifies the entire chassis.

### interface

Specifies an Ethernet interface.

#### fibrechannel

Specifies a fibrechannel interface.

#### fortygigabitethernet

Specifies a 40-Gbps interface.

#### gigabitethernet

Specifies a 1-Gbps interface.

#### hundredgigabitethernet

Specifies a 100-Gbps interface.

#### hundredgigabitethernet

Specifies a 100-Gbps interface.

*rbridge-id/slot/port*

Specifies the RBridge ID, slot, and port of the interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

The command **beacon enable chassis** has the same effect as the command **chassis beacon**, but the resulting state is reported for the latter.

## Examples

To disable the beacon for a chassis:

```
device# beacon chassis disable
device#
```

To enable the beacon on a 10-Gbps Ethernet interface for RBridge ID 1:

```
device# beacon enable interface te 1/0/1
device#
```

# bfd

Enables Bidirectional Forwarding Detection (BFD).

## Syntax

```
bfd
no bfd
```

## Command Default

BFD is disabled by default.

## Modes

- OSPF router configuration mode
- OSPFv3 router configuration mode
- OSPF router VRF configuration mode
- OSPFv3 router VRF configuration mode
- VXLAN overlay gateway site configuration mode

## Usage Guidelines

Use the **no** form of this command in OSPF router or OSPFv3 router mode to disable BFD globally. Use the **bfd** command in OSPF router configuration mode to create BFD sessions on all OSPFv2 interfaces on which BFD has been configured using the **ip ospf bfd** command. Use the **bfd** command in OSPFv3 router configuration mode to create BFD sessions on all OSPFv3 interfaces on which BFD has been configured using the **ipv6 ospf bfd** command.

Use the **bfd** command in VXLAN overlay gateway site configuration mode to configure BFD for L2 extension tunnels. Use the **no** form of this command in VXLAN overlay gateway site configuration mode to disable BFD for the tunnel.

## Examples

This example enables BFD globally in OSPF router configuration mode.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router ospf
device(config-router-ospf-vrf-default-vrf)# bfd
```

This example disables BFD globally in OSPFv3 router configuration mode.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# no bfd
```

This example enables BFD globally in a nondefault VRF instance.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf vrf red
device(config-ipv6-router-ospf-vrf-red)# bfd
```

This example enables BFD on a VXLAN overlay gateway site.

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# site s1
device(config-site-s1)# bfd
```

## History

Release version	Command history
6.0.1	This command was introduced.



# bfd holdover-interval

Sets the time interval for which OSPF or BGP routes are withdrawn after a BFD session is declared down.

## Syntax

`bfd holdover-interval time`

`no bfd holdover-interval time`

## Command Default

The BFD holdover interval is set to 0 by default.

## Parameters

*time*

Specifies BFD holdover interval in seconds. In BGP configuration mode, valid values range from 1 through 30 and the default is 0. In OSPF VRF and OSPFv3 VRF configuration mode, valid values range from 1 through 20, and the default is 0.

## Modes

BGP configuration mode.

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The BFD holdover interval is supported for both single-hop and multi hop sessions.

In BGP configuration mode, use this command to set the BFD holdover-time interval globally for BGP. In OSPF router configuration mode or OSPF router VRF configuration mode, use this command to set the BFD holdover-time interval globally for OSPFv2. In OSPFv3 router or OSPFv3 router VRF configuration modes, use this command to set the BFD holdover-time interval globally for OSPFv3.

The **no** form of the command removes the configured BFD holdover interval from the configuration, and reverts to the default value of 0.

## Examples

The following example sets the BFD holdover interval globally to 15 in BGP configuration mode..

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# bfd holdover-interval 15
```

The following example sets the BFD holdover interval globally to 12 in OSPF router configuration mode.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router ospf
device(config-router-ospf-vrf-default-vrf)# bfd holdover-interval 12
```

The following example sets the BFD holdover interval globally to 20 in OSPFv3 router configuration mode.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# bfd holdover-interval 20
```

The following example sets the BFD holdover interval globally to 20 for VRF instance "red" in OSPFv3 router VRF configuration mode.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf vrf red
device(config-ipv6-router-ospf-vrf-red)# bfd holdover-interval 20
```

## History

Release version	Command history
6.0.1	This command was introduced.

# bfd interval

Configures Bidirectional Forwarding Detection (BFD) session parameters on an interface or on a VXLAN overlay gateway site.

## Syntax

**bfd interval** *transmit-time* **min-rx** *receive-time* **multiplier** *number*

**no bfd interval** *transmit-time* **min-rx** *receive-time* **multiplier** *number*

## Command Default

Default parameters are used.

## Parameters

**interval** *transmit-time*

Specifies the interval, in milliseconds, a device waits to send a control packet to BFD peers. In interface subtype configuration mode and BGP configuration mode, valid values range from 50 through 30000, and the default is 500 on the VDX 6740, VDX 6740T, and VDX 6940 platforms. In interface subtype configuration mode and BGP configuration mode, valid values range from 50 through 30000, and the default is 200 on the VDX 8770 platforms. In VXLAN overlay gateway site configuration mode, valid values range from 100 through 30000, and the default is 100 on all platforms.

**min-rx** *receive-time*

Specifies the interval, in milliseconds, a device waits to receive a control packet from BFD peers. In interface subtype configuration mode and BGP configuration mode, valid values range from 50 through 30000, and the default is 500 on the VDX 6740, VDX 6740T, and VDX 6940 platforms. In interface subtype configuration mode and BGP configuration mode, valid values range from 50 through 30000, and the default is 200 on the VDX 8770 platforms. In VXLAN overlay gateway site configuration mode, valid values range from 300 through 30000, and the default is 300 on all platforms.

**multiplier** *number*

Specifies the number of consecutive BFD control packets that must be missed from a BFD peer before BFD determines that the connection to that peer is not operational. Valid values range from 3 through 50. The default is 3.

## Modes

BGP configuration mode

Interface subtype configuration mode

VXLAN overlay gateway site configuration mode

## Usage Guidelines

The **interval** *transmit-time* and **min-rx** *receive-time* variables are the intervals desired by the local device. The actual values in use will be the negotiated values.

Use the **bfd interval** command in BGP configuration mode for single-hop sessions only. Multihop sessions in BGP use either the values configured at interface level using the **bfd interval** command or the default interval values.

The **no** form of the command reverts to the default parameters.

## Examples

The following example sets the BFD session parameters globally for a 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/10
device(conf-if-fo-101/0/10)# bfd interval 100 min-rx 100 multiplier 10
```

The following example sets the BFD session parameters globally for a virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 24
device(config-ve-24)# bfd interval 120 min-rx 150 multiplier 8
```

The following example sets the BFD session parameters globally for BGP.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# bfd interval 140 min-rx 125 multiplier 44
```

The following example sets the BFD session parameters on a VXLAN overlay gateway site.

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# site s1
device(config-site-s1)# bfd interval 2000 min-rx 3000 multiplier 26
```

## History

Release version	Command history
6.0.1	This command was introduced.

# bfd shutdown

Disables Bidirectional Forwarding Detection (BFD) on an interface.

## Syntax

**bfd shutdown**

**no bfd shutdown**

## Command Default

This command is disabled by default.

## Modes

Interface subtype configuration mode.

## Usage Guidelines

Use the **no** form of this command to re-enable BFD sessions.

## Examples

This example disables BFD sessions on a specific 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/10
device(config-if-fo-101/0/10)# bfd shutdown
```

This example disables BFD sessions on a specific virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 24
device(config-ve-24)# bfd shutdown
```

## History

Release version	Command history
6.0.1	This command was introduced.

# bfd-session-setup-delay

Provides a time delay before establishing the single hop BFD session after the port is enabled.

## Syntax

**bfd-session-setup-delay** *seconds*

**no bfd-session-setup-delay** *seconds*

## Command Default

By default, the time delay to establish the single hop session is set to 180 seconds.

## Parameters

*seconds*

The time delay in seconds. You can specify a value between 5 and 600 seconds. The default value is 180 seconds.

## Modes

Rbridge-ID configuration mode

## Usage Guidelines

The **no** form of the command removes the time delay for the session.

## Examples

The following example sets a delay time of 40 seconds before establishing the single hop session.

```
device# configure terminal
device(config)# rbridge-id 11
device (config-rbridge-id-11)# bfd sh-session-setup-delay 40
```

## History

Release version	Command history
7.0.0	This command was introduced.

# bfd-shutdown

Shuts down the Bidirectional Forwarding Detection (BFD) daemon.

## Syntax

**bfd-shutdown**

**no bfd-shutdown**

## Modes

RBridge ID configuration mode

## Usage Guidelines

By default, the BFD daemon is enabled.

Run the **bfd-shutdown** command to shut down the BFD daemon

To restore normal BFD functionality (after running the **bfd-shutdown** command), run the **no bfd-shutdown** command and restart the system. BFD functionality is only available again after the system is restarted.

## Examples

The following example shuts down the BFD daemon.

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# bfd-shutdown
```

## History

Release version	Command history
7.2.0c	This command was introduced.

# bgp-redistribute-internal

Causes the device to allow the redistribution of IBGP routes from BGP into OSPF for non-default VRF instances.

## Syntax

**bgp-redistribute-internal**

**no bgp-redistribute-internal**

## Command Default

This feature is disabled.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

By default, with default VRF instances, the device does not allow the redistribution of IBGP routes from BGP4 and BGP4+ into OSPF. This helps to eliminate routing loops. In non-default VRF instances, use this command to allow the redistribution of IBGP routes from BGP into OSPF. This command is enabled only if a non-default VRF instance has been specified.

## Examples

This example enables BGP4 route redistribution.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# bgp-redistribute-internal
```

This example enables BGP4+ route redistribution for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# bgp-redistribute-internal
```



## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# bind

Creates a persistent binding between the logical FCoE port and the ten/forty gigabit or LAG port.

## Syntax

```
bind { <N>gigabitethernet rbridge-id/slot/port | port-channel number | mac-address address }
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port).

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel interface.

### mac-address *address*

Specifies a MAC address. The valid format is HH:HH:HH:HH:HH:HH.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The configuration will be stored in the configuration and retained across reboots.

When the FCoE logical port is automatically bound to a TE/FO LAG port, it is referred as dynamic binding. This binding is valid only till the FLOGI session is valid. The binding is automatically removed when CNA logs out.

### ATTENTION

Only one type of binding can be used for each physical port, so the ten/forty gigabit Ethernet or LAG (MAC) binding configurations overwrite each other.

## Examples

```
device# configure terminal
device(config)# interface fcoe 1/1/55
device(config-Fcoe-1/1/55)# bind tengigabitethernet 1/0/1
device(config)# interface fcoe 1/1/56
device(config-Fcoe-1/1/56)# bind mac-address 00:05:1e:c5:96:a4
```

# bp-rate-limit heavy module

Configures the blade processor (BP) rate limit as "heavy" for one or more slots or modules, helping to reduce the amount of trapped traffic.

## Syntax

```
bp-rate-limit heavy module { add slot/module | remove slot/module }
```

## Command Default

For slots to which this command is not applied, the BP rate limit is the normal limit.

## Parameters

### add

Adds a slot or module or range of slots or modules. Comma delimiters and ranging are allowed.

### remove

Removes a slot or module or range of slots or modules. Comma delimiters and ranging are allowed.

*slot/module*

A slot or module number from 0 through 16. A Top-of-Rack (ToR) device ("pizzabox") is indicated by "0".

## Modes

RBridge ID configuration mode

## Usage Guidelines

There is no "no" form of this command. To restore the default (normal) BP rate limit for one or more slots or modules, use the **remove** keyword.

You can combine nonconsecutive slots as well as ranges of slots or modules.

## Examples

The following example applies the heavy BP rate limit to slot 1.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# bp-rate-limit heavy module add 1
```

The following example applies the heavy BP rate limit to slots 1, 4 through 7, and 9.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# bp-rate-limit heavy module add 1,4-7,9
```

The following example applies the heavy BP rate limit to a ToR device.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# bp-rate-limit heavy module add 0
```

The following example restores the default BP rate limit to a range of slots or modules.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# bp-rate-limit heavy module remove 4-7
```

## History

Release version	Command history
6.0.0	This command was consolidated and updated.
6.0.1a	This command was updated with the <b>add</b> and <b>remove</b> keywords.

# bpdu-drop enable

Drops STP, RSTP, MSTP, and PVST and RPVST bridge protocol data units (BPDUs), disabling the tunneling of those protocols on an interface.

## Syntax

```
bpdu-drop enable [ rx | tx | all ]
```

```
no bpdu-drop enable [ rx | tx | all ]
```

## Command Default

BPDU-drop is disabled.

## Parameters

- tx** Disables tunneling in the transmit direction.
- rx** Disables tunneling in the receive direction.
- all** Disables tunneling in both the transmit and receive directions.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command prevent reception of any STP or PVST BPDUs on a interface. If such a BPDU is received on a interface that is BPDU-drop enabled, the interface drops the BPDU frames, but does not shut down.

The **bpdu-drop enable all** command may only be added to physical interfaces and is not supported on Port-Channel interfaces. When the command is added to a Port-Channel, it does not error out but it does not drop BPDUs either.

Enter **bpdu-drop enable** with the **tx**, **rx**, or **all** options. Without an optional keyword, the action applies to the ingress direction only.

Enter **no bpdu-drop enable** with the **tx**, **rx**, or **all** options to disable BPDU drop in one or more directions.

Enter **no bpdu-drop enable** to disable BPDU-drop completely.

## Examples

To enable BPDU-drop on a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet178/0/9
device(config-if-te-178/0/9)# bpdu-drop enable
```

To disable BPDU-drop on a specific port-channel interface:

```
device(config)# interface port-channel 62
device(config-Port-channel-62)# no bpdu-drop enable
```

To disable BPDU-drop on a specific port-channel interface in the transmit direction:

```
device(config)# interface port-channel 62
device(config-Port-channel-62)# no bpdu-drop enable tx
```

## Related Commands

[interface](#)

# bridge-priority

Specifies the bridge priority for the common instance.

## Syntax

**bridge-priority** *priority*

**no bridge-priority**

## Command Default

The default priority is 32768.

## Parameters

*priority*

Specifies the bridge priority. Valid values range from 0 through 61440 in increments of 4096.

## Modes

Protocol Spanning Tree mode

## Usage Guidelines

The priority values can be set only in increments of 4096.

Using a lower priority value indicates that the bridge might become root.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

Enter **no bridge-priority** to return to the default priority.

## Examples

To specify the bridge priority:

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# bridge-priority 8192
```

```
device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# bridge-priority 8192
```

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# bridge-priority 8192
```

## Related Commands

[protocol spanning-tree](#)



## bsr-candidate

Configures a bootstrap router (BSR) as a candidate to distribute rendezvous point (RP) information to the other PIM Sparse devices within a PIM Sparse domain.

### Syntax

```
bsr-candidate interface <N>gigabitethernet [hash-mask-length] [priority]  
no bsr-candidate
```

### Command Default

The PIM router does not participate in BSR election.

### Parameters

**interface** <N>*gigabitethernet*

Specifies the interface type being configured as a BSR candidate.

*hash-mask-length*

Specifies the hash mask length used in BSR messages. The range is from 0 to 32. The default is 30.

*priority*

Specifies the BSR priority. The range is from 0 to 255, from low to high. The default is 64.

### Modes

PIM Router configuration mode

### Usage Guidelines

The **no** form of this command makes the PIM router cease to act as a candidate BSR.

Each PIM Sparse domain has one active BSR. For redundancy, you can configure ports on multiple devices as candidate BSRs. The PIM Sparse protocol uses an election process to select one of the candidate BSRs as the BSR for the domain. The BSR with the highest BSR priority is elected. If the priorities result in a tie, the candidate BSR interface with the highest IP address is elected.

Although you can configure the device as only a candidate BSR or an RP, it is recommended that you configure the same interface on the same device as both a BSR and an RP.

### Examples

The following example uses a tengigabit ethernet interface to configure a device as a candidate BSR.

```
device(config)# rbridge 1  
device(config-rbridge-id-1)# router-pim  
device(config-pim-router)# bsr-candidate interface tengigabitethernet 1/1/2 hash-mask-length 32 bsr-  
priority 10
```

## History

Release version	Command history
7.1.0	This command was introduced.

# bsr-msg-interval

Sets the PIM BSR message interval timer.

## Syntax

```
bsr-msg-interval time
no bsr-msg-interval time
```

## Command Default

The default IPv4 PIM BSR message interval timer is 60 seconds.

## Parameters

*time*

Defines the interval at which the BSR sends RP candidate data to all IPv4-enabled routers within the PIM Sparse domain. Valid values are 10 to 65535 seconds. The default is 60 seconds.

## Usage Guidelines

The BSR message interval timer defines the interval at which the BSR sends RP candidate data to all routers within the PIM Sparse domain.

## Examples

The following example sets the PIM BSR message interval timer to 30 seconds.

```
device(config)# router pim
device(conf-pim-router)# bsr-msg-interval 30
```

## History

Release version	Command history
7.1.0	This command was introduced.

# capability as4-enable

Enables 4-byte autonomous system number (ASN) capability at the BGP global level.

## Syntax

`capability as4-enable`

`no capability`

## Command Default

This feature is disabled.

## Modes

BGP configuration mode

## Usage Guidelines

Use the **no** form of this command to disable this functionality.

## Examples

The following example enables 4-byte ASN capability:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# capability as4-enable
```

# capture packet interface

Enables the capture of packet information on an interface, for display on the switch itself or for storage in an automatically generated file.

## Syntax

```
capture packet { interface } { all | <N>gigabitethernet rbridge-id/slot/port | port-channel number | ve vlan_id } { direction { both | rx | tx } } { filter { I2 | I3 | all }
```

## Parameters

### interface

Selects an interface (required).

### all

Selects all interfaces.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *N*gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### port-channel *number*

Specifies the port-channel number. The number of available channels ranges from 1 through 6144.

### ve *vlan\_id*

Specifies the virtual Ethernet interface. Range is from 1 through 8191.

### direction

Selects a direction (required).

#### both

Selects traffic in both transmit and receive directions.

#### rx

Selects received traffic.

#### tx

Selects transmitted traffic.

### filter

Selects the packet types to be filtered (required).

#### I2

Filters only Layer 2 packets to the CPU.

**I3**

Filters only Layer 3 packets to the CPU.

**all**

Filters all packets to the CPU, including transit packets if an access control list (ACL) is enabled (Refer to the Usage Guidelines.)

## Modes

Privileged EXEC mode

## Usage Guidelines

Capturing packet information on an interface can provide significant help in debugging, especially for Layer 2 TRILL and Layer 3 packets. Captured packets are stored in a circular buffer, and they are also written to an automatically generated "pktcapture.pcap" file, which can store up to 1500 K of data in flash memory (the equivalent of approximately 10k packets, each having an average size of 100 bytes). Once this file is full, it is saved at \*\_old.pcap and data are written to a new pktcapture.pcap file. These files can be exported and viewed through a packet analyzer such as Wireshark.

**NOTE**

Up to 100 packets per interface can be captured. Once the buffer is filled, the oldest packets are replaced with the most recent.

This command can be entered on any RBridge in an Extreme data center fabric.

To disable packet capture globally, use the **no capture packet all** command.

**NOTE**

The **all** option is not supported for enabling packet capture.

To view the captured information on the switch, use the **show capture packet interface** command.

Note the following limitations:

- Support is provided only on physical (Ethernet) interfaces, not on logical interfaces. To see packets on logical interfaces, first enable the capture on the corresponding physical interfaces.
- In the initial release, support for capturing transit traffic requires ACL logging.
- Packets that are dropped in the ASIC cannot be captured.

**CAUTION**

**Capturing packets over multiple sessions and over long durations can affect system performance.**

## Examples

To capture received packets on a VE interface:

```
device# capture packet interface ve 20 direction rx
```

To capture received packets on a VE interface:

```
device# capture packet interface ve 20 direction rx
```

## History

Release version	Command history
7.1.0	This command was modified to support port-channel and VE interfaces.

## Related Commands

[show capture packet interface](#)

# cbs

Mandatory command for configuring the controlled burst size for a class-map.

## Syntax

**cbs** *cbs-size*

**no cbs** *cbs-size*

## Parameters

*cbs-size*

Controlled burst size. Valid values range from 1250 through 5000000000 bytes in increments of 1 byte. This is a mandatory parameter for configuring a class-map.

## Modes

Policy map class police (*config-policymap-class-police*) configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class map.

If the optional parameters for a class-map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on the VDX 8770-4, VDX 8770-8, and later switches.

Use the **no** version of this command to remove the parameter from the class-map.

## Examples

This example configures a class-map called "default" within a policy-map.

```
device# configure terminal
device(config)# policy-map policymap1
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# cbs 50000
```



## cee

Applies a Converged Enhanced Ethernet (CEE) provisioning map on an interface.

## Syntax

`cee default`

`no cee`

## Command Default

There is no CEE provisioning applied on an interface. The only map name allowed is "default."

## Modes

Interface subtype configuration mode

## Usage Guidelines

The CEE map applied on an interface should already exist on the switch.

Enter `no cee` to remove the CEE provisioning map.

## Examples

To apply a CEE map to a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# cee default
```

# cee-map (configuration)

Enters the CEE map configuration mode.

## Syntax

**cee-map default**

## Command Default

The only map name allowed is "default."

## Modes

Global configuration mode

## Usage Guidelines

Only a single CEE map is allowed, named "default." It is created when system starts up. The initial configuration of the default CEE map is:

```
Precedence 1
Priority Group Table
 1: Weight 40, PFC Enabled, BW% 40
 2: Weight 60, PFC Disabled, BW% 60
15.0: PFC Disabled
15.1: PFC Disabled
15.2: PFC Disabled
15.3: PFC Disabled
15.4: PFC Disabled
15.5: PFC Disabled
15.6: PFC Disabled
15.7: PFC Disabled
Priority Table
  CoS:   0   1   2   3   4   5   6   7
-----
 PGID:  2   2   2   1   2   2   2   2
Enabled on the following interfaces
```

## Related Commands

[cee](#), [fcoeport](#), [priority-group-table](#)

# certutil import ldapca

Imports a Lightweight Directory Access Protocol (LDAP) Certification Authority (CA) certificate from a remote server

## Syntax

```
certutil import ldapca directory path file filename protocol { FTP | SCP } host remote_ip user user_acct password password
ssh
```

```
certutil import ldapca | syslogca directory ca certificatepath protocol { FTP | SCP } host remote_ip user user_acct password
password [ rbridge-id { rbridge_id | all } ]
```

```
no certutil ldapca [ rbridge-id { rbridge-id | all } ]
```

## Parameters

**directory** *path*

Specifies the path to the certificate.

**file** *filename*

Specifies the filename for the certificate.

**host** *remote\_ip*

Specifies the IP address of the remote host.

**password** *password*

Specifies the password to access the remote host.

**protocol** *FTP* | *SCP*

Specifies the protocol used to access the remote server.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**user** *user\_acct*

Specifies the user name to access the remote host.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command supports FTP and SCP.

Enter **no certutil ldapca** to delete the LDAP CA certificates of all Active Directory (AD) servers.

## Examples

To import the SSH public key for user "admin" from the remote host:

```
switch# certutil import sshkey user admin host 10.70.4.106 directory /users/home40/bmeenaks/.ssh file
id_rsa.pub login fvt
Password: *****
switch# 2012/11/14-10:28:58, [SEC-3050], 75,, INFO, VDX6720-60, Event: sshutil, Status: success, Info:
Imported SSH public key from 10.70.4.106 for user 'admin'.
switch#
```

To delete all LDAP CA certificates:

```
switch# no certutil ldapca
```

In VCS mode

To import a certificate for LDAP:

```
switch# certutil import ldapca directory /usr/ldapcert/ file cacert.pem protocol SCP host 10.23.24.56
user admin password rbridge-id 3
password:
switch#
```

To delete LDAP certificates on rbridge-id 3:

```
switch# no certutil syslogca rbridge-id 3
Do you want to delete syslogca certificate? [y/n]:y
Warning: All the syslog CA certificates are deleted.
switch
```

# certutil import sshkey

Imports the SSH public key for an SSH user from the remote host using the mentioned login credentials and path name.

## Syntax

```
certutil import sshkey host remote_ip_address directory ssh_public_key_path user user_acct password password login
  login_id [ rbridge-id { rbridge-id | all } ]
no certutil sshkey [ rbridge-id { rbridge-id | all } ]
```

## Parameters

**directory** *path*

Specifies the path to the certificate.

**file** *filename*

Specifies the SSH public key with a .pub extension.

**host** *remote\_ip*

Specifies the IP address of the remote host.

**login** *login\_id*

Specifies the login name in the remote host.

**password** *password*

Specifies the password to access the remote host.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**user** *user\_acct*

Specifies the user name to access the remote host.

## Modes

Privileged EXEC mode

## Usage Guidelines

Enter **no certutil sshkey user** to delete the SSH public key a specified user.

When using the 'pass' parameter with special characters (such as #,\$@`) use single or double-quotes around the password. Alternatively, the special characters can be escaped with a backslash (\) preceding the special character.

## Examples

In VCS mode

The following command imports a public CA certificate:

```
device# certutil import sshkey user admin host 10.70.4.106 directory /users/home40/bmeenaks/.ssh file
id_rsa.pub login fvt rbridge-id 3
Password: *****
device# 2012/11/14-10:28:58, [SEC-3050], 75,, INFO, VDX6720-60, Event: sshutil, Status: success, Info:
Imported SSH public key from 10.70.4.106 for user 'admin'.
```

The following command deletes the SSH public key for "testuser."

```
device# no certutil sshkey user testuser rbridge-id 3
Do you want to delete the SSH public key file? [y/n]:y
device# 2012/11/11-13:46:05, [SEC-3050], 3295,, INFO, VDX6720-24, Event: sshutil, Status: success,
Info: Deleted SSH public keys associated to user 'testuser'.
```

The following command deletes the SSH public key for "testuser."

```
device# no certutil sshkey user testuser
Do you want to delete the SSH public key file? [y/n]:y
device# 2012/11/11-13:46:05, [SEC-3050], 3295,, INFO, VDX6720-24, Event: sshutil, Status: success,
Info: Deleted SSH public keys associated to user 'testuser'.
```

The following commands demonstrate the use of special characters in a password.

```
device# certutil import ssh host 192.168.10.10 dir /home/brcd1/.ssh file id_rsa.pub user admin login
brcd1 pass Abcde\!
device# certutil import ssh host 192.168.10.10 dir /home/brcd1/.ssh file id_rsa.pub user admin login
brcd1 pass "Abcde!"
```

## Related Commands

[show cert-util sshkey](#)

# certutil import syslogca

Imports a syslog CA certificate.

## Syntax

```
certutil import syslogca directory path file filename protocol { FTP | SCP } host remote_ip user user_acct password
password [ rbridge-id { rbridge-id | all } ]
```

```
no certutil syslogca
```

## Parameters

**directory** *path*

Specifies the path to the certificate.

**file** *filename*

Specifies the filename for the certificate.

**host** *remote\_ip*

Specifies the IP address of the remote host.

**password** *password*

Specifies the password to access the remote host.

**protocol** *FTP | SCP*

Specifies the protocol used to access the remote server.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**user** *user\_acct*

Specifies the user name to access the remote host.

## Modes

Privileged EXEC mode

## Usage Guidelines

Enter **no certutil syslogca** to delete the syslog CA certificates of all Active Directory servers.

## Examples

The following command deletes a syslog CA certificate:

```
switch# no certutil syslogca  
Do you want to delete syslog CA certificate? [y/n]:
```

In VCS mode

The following command imports a syslog CA certificate:

```
switch# certutil import syslogca directory /usr/ldapcert/ file cacert.pem protocol SCP host  
10.23.24.56 user admin password rbridge-id 3  
password:  
switch#
```

The following command deletes a syslog CA certificate:

```
switch# no certutil syslogca rbridge-id 5  
Do you want to delete syslog CA certificate? [y/n]:
```



# channel-group

Enables Link Aggregation on an interface.

## Syntax

```
channel-group number mode { active | passive | on } [ type { standard | extreme } ]
no channel-group
```

## Command Default

The value for **type** is set to **standard**.

## Parameters

*number*

Specifies a Link Aggregation Group (LAG) port channel-group number to which this link should administratively belong to. Valid values range from 1 through 6144.

**mode**

Specifies the mode of Link Aggregation.

**active**

Enables the initiation of LACP negotiation on an interface.

**passive**

Disables LACP on an interface.

**on**

Enables static link aggregation on an interface.

**type**

Specifies the type of LAG.

**standard**

Specifies the 802.3ad standard-based LAG.

**extreme**

Specifies the Extreme proprietary hardware-based trunking.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command adds an interface to a port-channel specified by the channel-group number. This command enables link aggregation on an interface, so that it may be selected for aggregation by the local system.

The maximum number of LAGs varies, depending on what Extreme hardware you have installed:

**TABLE 5** Maximum number of LAGs (Port Channels) per hardware model

VDX 6740, 6740T, 6740T-1G	VDX 8770	VDX 6940-36Q	VDX 6940-144S
64	288	144	144

Be aware of the following:

- A maximum of four link aggregation groups can be created per switch when the **type** is set to **extreme**.
- A maximum of four links can become part of a single aggregation group when the **type** is set to **extreme** and they must be on the same port-channel.
- Links 0 through 7 belong to port-channel 1; links 8 through 15 belong to port-channel 2, and links 16 through 23 belong to port-channel 3.
- For the **standard** type, a maximum of 16 links can be aggregated per aggregation group and they can be members of any port-channel.
- Enter **no channel-group** to remove the port-channel members.

**NOTE**

For additional discussion of using the **channel-group** command, refer to the section "Configuring FCoE" in the *Extreme Network OS Layer 2 Switching Configuration Guide*.

## Examples

To set the channel-group number to 4 and the mode to *active* on a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# channel-group 4 mode active
```

To set the channel-group number to 10, the mode to *passive*, and the type to *extreme* on a specific 1-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface gigabitethernet 170/0/1
device(conf-if-gi-170/0/1)# channel-group 10 mode passive extreme
```

## History

Release version	Command history
5.0.1	This command was updated with a reference to the <i>Extreme Network OS Layer 2 Switching Configuration Guide</i> .

## Related Commands

[interface](#)

# chassis

Sets the IPv4 or IPv6 address of a device chassis.

## Syntax

```
chassis { virtual-ip IPv4-address | virtual-ipv6 IPv6-address }
no chassis { virtual-ip | virtual-ipv6 }
```

## Command Default

The default is the initial address of the device chassis.

## Parameters

**virtual-ip** *IPv4-address*

Sets an IPv4 address in dotted-decimal notation with a CIDR prefix (mask).

**virtual-ipv6** *IPv6-address*

Sets an IPv6 address in colon-separated hexadecimal notation with a CIDR prefix.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This command changes the default chassis IPv4 or IPv6 address. The default is the initial address of the device chassis.

This is the address that is used to access devices through their RBridge ID.

Use this command to change the IP address to facilitate management, for example, if a device is moved to a different subnet. The IP address of the management platform should be in the same subnet as the devices it manages.

This command applies only to chassis switches, for example, the VDX 8770.

Use the **no** form of this command to revert to the default address.

## Examples

IPv4:

```
device# configure terminal
device(config)# rbridge-id 4
device(config-rbridge-id-4)# chassis virtual-ip 10.11.12.13/20
```

IPv6:

```
device# configure terminal
device(config)# rbridge-id 4
device(config-rbridge-id-4)# chassis virtual-ipv6 2001:db8:8086:6502/64
```

## Related Commands

[show rbridge-id](#), [show running-config](#)

# chassis beacon

Enables the flashing LED beacon on the switch chassis, which makes it easier to find the specified chassis in large data centers.

## Syntax

```
chassis beacon { enable | disable }
```

## Command Default

The LED is disabled.

## Parameters

### enable

Enables the chassis beacon LED.

### disable

Disables the chassis beacon LED.

## Modes

Privileged EXEC mode

## Usage Guidelines

The command **chassis beacon** has the same effect as the command **beacon enable chassis**, but the resulting state is not reported for the latter.

## Examples

To enable the chassis beacon:

```
device# chassis beacon enable
Chassis Beacon has been enabled
```

To disable the chassis beacon:

```
device# chassis beacon disable
Chassis Beacon has been disabled
```

# chassis disable

Disables all interfaces in the chassis.

## Syntax

**chassis disable**

## Modes

Privileged EXEC mode

## Usage Guidelines

All interfaces will be taken offline

This command is supported only on the local switch.

Enter **chassis disable** before making configuration changes or running offline diagnostics.

You must execute the **chassis enable** command after running offline diagnostics, or the switch will not boot correctly.

## Examples

To disable all interfaces on the local switch:

```
device# chassis disable
```

## Related Commands

[chassis enable](#)

# chassis enable

Enables all interfaces in the chassis.

## Syntax

```
chassis enable
```

## Modes

Privileged EXEC mode

## Usage Guidelines

All interfaces that passed the power-on self-test (POST) are enabled. They may come online if connected to a device, or remain offline if disconnected. Enter **chassis enable** to re-enable the chassis after making configuration changes or running offline diagnostics.

This command is supported only on the local switch.

You must execute the **chassis enable** command after running offline diagnostics, or the switch will not boot correctly.

## Examples

To enable all interfaces on the local switch:

```
device# chassis enable
```

## Related Commands

[chassis disable](#)

# chassis fan airflow-direction

Specifies the direction of airflow through the chassis based on physical PSU and fans.

## Syntax

```
chassis fan airflow-direction [ port-side-intake | port-side-exhaust ]
```

## Parameters

### port-side-intake

Specifies the airflow to enter the switch.

### port-side-exhaust

Specifies the airflow to exit the switch.

## Modes

Global configuration mode

## Usage Guidelines

This command must only be used after you purchase and install the appropriate fan/power supply that provides the desired airflow direction in the switch. Please contact your Sales Representative to obtain the correct part numbers and pricing.

When the **chassis fan airflow-direction** command is issued, the switch will not recognize the configuration change until the switch is rebooted.

Only one (1) configuration change is accepted per reboot. This means that even if this command is entered multiple times, only the first configuration change entered will be effective after rebooting.

The switch serial number is registered with Extreme and the information recorded in the Extreme database about that switch includes the airflow orientation at the time of shipment. Any subsequent change in airflow direction is not recorded in the Extreme database. This means that if you request a Return Merchandise Authentication (RMA) for the switch, the replacement switch will be sent with the original orientation.

## Examples

To specify the fan airflow-direction:

```
device# chassis fan airflow-direction port-side-exhaust
```

```
Previous configuration : port-side-intake
Current configuration  : port-side-exhaust
System fan airflow-direction changes will be effective after reboot!!
```



# chassis power-cycle-db-shutdown

Shuts down the chassis configuration database, without rebooting the device, and disconnects the node from the rest of the cluster.

## Syntax

```
chassis power-cycle-db-shutdown
```

## Command Default

The chassis configuration database is running normally.

## Modes

Privileged EXEC mode

## Usage Guidelines

When devices encounter abrupt power cycles, there have been rare cases of device configuration database corruption. This database corruption causes the device to reboot and reverts the device to the default configuration if it is not part of the cluster. In the case of scheduled power-cycles, it is recommended to use the **chassis power-cycle-db-shutdown** command before actually rebooting or power-cycling the device.

### ATTENTION

This command should not be executed for planned upgrades. As switches are rebooted gracefully during upgrades, there is no need to shut down the database. In fact, shutting down the database will cause an upgrade to fail.

The rest of the cluster is informed that the node is to be power-cycled and removed from the cluster. All commands (except for the **reload** command) are blocked on this node until the node is rebooted or power-cycled. Cluster formation requests to this node are ignored. The node is not fully functional until it reboots or is power-cycled.

## Examples

Typical command execution.

```
device# chassis power-cycle-db-shutdown
```

## History

Release version	Command history
7.1.0	This command was introduced.

# cidrecov

Recovers data from Chassis ID cards if possible.

## Syntax

`cidrecov`

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command if you receive an error or warning RASLog message that instructs you to run this command.

Two chassis ID (CID) cards contain data necessary for system operation. Each CID contains two Serial Electronically Erasable Programmable Read Only Memory (SEEPROM) devices. If data on either card becomes corrupt or mismatched, a regularly run CID audit writes messages to the RASLog. Follow the instructions in the messages. Mismatched data can be reset, and corrupt data can sometimes be recovered if the corrupt data is on the non-critical SEEPROM.

This command is supported only on the VDX 8770-4 and VDX 8770-8 switches.

## Examples

Example 1: Noncritical SEEPROM is inaccessible or corrupt, but recovery becomes possible:

```
device# cidrecov

CID 1 Non-Critical Seeprom is Inaccessible or Corrupted.
  CID Non-Critical Seeprom Problem Details
CID 1 Non-Critical Seeprom IP address Control Data Checksum Bad !!!!
CID 1 IP address Control Data:
Version: 0xa
Checksum: 0x0
Size: 0x3
CID 2 IP address Control Data:
Version: 0xa
Checksum: 0x7
Size: 0x3
***WARNING: Recovering IP Data May Affect Both IP Control and IP Records ***
Backup Current Data Displayed Below If Needed.
CID 1 Chassis Name: VDX8770-4
CID 2 Chassis Name: VDX8770-4
CID 1 IP address Control Data:
Version: 0xa
Checksum: 0x0
Size: 0x3
CID 2 IP address Control Data:
Version: 0xa
Checksum: 0x7
Size: 0x3
IP address Record 1 on CID 1
1st IP Address: 10.17.19.53
1st IP Mask: 255.255.240.0
2nd IP Address: 10.17.19.54
2nd IP Mask: 255.255.240.0
Gateway Address: 10.17.16.1
IP address Record 1 on CID 2
1st IP Address: 10.17.19.53
1st IP Mask: 255.255.240.0
2nd IP Address: 10.17.19.54
2nd IP Mask: 255.255.240.0
Gateway Address: 10.17.16.1
IP address Record 2 on CID 1
1st IP Address: 10.17.19.52
1st IP Mask: 255.255.240.0
2nd IP Address: 0.0.0.0
2nd IP Mask: 0.0.0.0
Gateway Address: 0.0.0.0
IP address Record 2 on CID 2
1st IP Address: 10.17.19.52
1st IP Mask: 255.255.240.0
2nd IP Address: 0.0.0.0
2nd IP Mask: 0.0.0.0
Gateway Address: 0.0.0.0
  CID Recovery Options
0. Exit
1. Recover with default values
2. Recover BAD from GOOD
Enter Selection > 2

Copy IP Data table...
  Copy 384 bytes from CID 2 to CID 1, num blks 1 resid 128
  Read block 1 from CID 2 succeeded
  Write block 1 to CID 1 succeeded
  Read last block from CID 2 succeeded
  Write last block to CID 1 succeeded
  copy successful
Copy succeeded for all data types attempted
IP Address CID Recovery completed.
```

Example 2: Non-critical SEEPROM is inaccessible or corrupt, but recovery is not possible:

```
device# cidrecov

CID 1 Non-Critical Seeprom is Inaccessible or Corrupted.
  CID Non-Critical Seeprom Problem Details
CID 1 Non-Critical Seeprom Read Failed.
Recovery is not possible. Please contact Extreme Technical Support for replacement of the inaccessible
CID(s).
```

Example 3: Critical SEEPROM data is mismatched, recovery is not possible:

```
device# cidrecov

CID 1 and CID 2 Critical Seeprom Data is Mismatched.
  CID Seeprom Problem Details
CID Seeprom Chassis Serial Number Mismatch.
CID 1 Serial Number: BYP3G15G00N
CID 2 Serial Number: BYP3G17H00P
Recovery is not possible. Please contact Extreme Technical Support for replacement of the corrupted
CID(s).
```

# cipherset

Configures FIPS-compliant ciphers for the Lightweight Directory Access Protocol (LDAP).

## Syntax

```
cipherset { ldap | radius }
```

## Command Default

There are no restrictions on LDAP ciphers.

## Parameters

*radius*

Specifies secure RADIUS ciphers.

*ldap*

Specifies secure LDAP ciphers.

## Modes

Privileged EXEC mode

## Usage Guidelines

A switch must be configured with secure ciphers for SSH before that switch can be FIPS compliant. If LDAP authentication is to be used, the LDAP ciphers are also required before a switch can be FIPS compliant.

The secure LDAP ciphers are EAS128-SHA and DES-CBC3-SHA.

This command can be entered only from a user account with the admin role assigned.

### NOTE

The **cipherset ssh** command has been deprecated in Network OS v6.0.1. Use the **ssh client cipher** or the **ssh server cipher** commands to set the SSH client's cipher lists for SSH clients and servers.

## Examples

This example configures secure RADIUS ciphers.

```
device# cipherset radius
RADIUS cipher list configured successfully
```

This example configures secure LDAP ciphers.

```
device# cipherset ldap
ldap cipher list configured successfully
```

## History

Release version	Command history
6.0.1	The <b>ssh</b> keyword was deprecated.
7.1.0	The <b>EAS256-SHA</b> cipher has been deprecated.

# cisco-interoperability

Configures the device to interoperate with some legacy Cisco switches.

## Syntax

```
cisco-interoperability { disable | enable }
```

## Command Default

Cisco interoperability is disabled.

## Parameters

### disable

Disables Cisco interoperability for the Multiple Spanning Tree Protocol (MSTP) device.

### enable

Enables Cisco interoperability for the MSTP enabled device.

## Modes

Protocol Spanning Tree MSTP mode

## Usage Guidelines

For some devices, the MSTP field, Version 3 Length, does not adhere to the current standards.

If Cisco interoperability is required on any device in the network, then all devices in the network must be compatible, and therefore enabled using this command for interoperability with a Cisco switch.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

## Examples

To enable Cisco interoperability on a device:

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# cisco-interoperability enable
```

To disable Cisco interoperability on a device:

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# cisco-interoperability disable
```

# class

Creates a class map in a policy map and enters the class map configuration mode.

## Syntax

```
class class-mapname
```

```
no class class-mapname
```

## Command Default

A policy map is not created.

## Parameters

*class-mapname*

The designated name for the class map.

## Modes

Policy map configuration mode

## Usage Guidelines

Use this command to configure a class map for a police policy map with QoS and policing parameters for inbound or outbound traffic. The class map must be previously created and associated with match criteria using the **class-map** command. (Refer to the **qos cos** command.)

When you enter the **class** command and access policy-map class configuration mode, you can configure QoS and policing parameters for the class map using the commands for the specific parameters.

The commands that set the parameters for a class map are:

- **cbs**
- **eir**
- **ebs**
- **conform-set-dscp**
- **conform-set-prec**
- **conform-set-tc**
- **exceed-set-dscp**
- **exceed-set-prec**
- **exceed-set-dscp**
- **police cir**
- **set-priority**

The QoS and policing parameters define the CIR, CBS, EIR, and EBS rates and the actions that must occur when traffic conforms or exceeds designated rates.



Each policy map can contain one class map.

Enter **no police** while in config-policymap-class mode to remove all policing parameters for the class map.

Enter **no police** command followed by a policing parameter name to remove a specific parameter.

#### NOTE

The **cir** and **cbs** parameters are mandatory for configuring a class map. Other parameters are optional. If optional parameters are not set then they will be treated as disabled. To delete the mandatory CIR or CBS parameters, you must delete all policer parameters while in the policy map class configuration mode using the **no police** command.

#### NOTE

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

## Examples

This example configures a class-map called "default" within a policy-map.

```
device# configure terminal
device(config)# policy-map policymap1
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# cbs 50000
device(config-policymap-class-police)# eir 800000
device(config-policymap-class-police)# ebs 400000
device(config-policymap-class-police)# conform-set-tc 3
device(config-policymap-class-police)# exceed-set-prec 4
```

# class-map

Enters class (classification) map configuration mode.

## Syntax

```
class-map class-map-name
```

```
no class-map class-map-name
```

## Command Default

The class name "class-default" is reserved and cannot be created by users.

## Parameters

*class-map-name*

Name of classification map. The map name is restricted to 64 characters.

## Modes

Global configuration mode.

## Usage Guidelines

The class map created using the **class-map** command becomes the default class map and cannot be removed using the **no class-map** command. You can remove a class map from a policy map however.

You can create up to 128 class maps.

### NOTE

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

## Examples

To create a class map and place system into config-classmap mode:

```
device(config)# class-map default  
device(config-classmap)#
```

# clear ag nport-utilization

Clears Access Gateway N\_Port utilization information.

## Syntax

```
clear ag nport-utilization [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

You can clear N\_Port utilization information either for a specific RBridge or for all.

#### *rbridge-id*

Specify an RBridge ID.

### all

Clear N\_Port utilization information for N\_Ports on all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Enabling a port also clears the information

There is a **show** form of this command, which shows Access Gateway N\_Port utilization information.

## Examples

The following command clears utilization information for Access Gateway N\_Ports on RBridge 1:

```
device# clear ag nport-utilization rbridge-ID 1
```

The following command clears Access Gateway utilization information for all N\_Ports on the switch:

```
device# clear ag nport-utilization rbridge-ID all
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[ag](#), [enable \(Access Gateway\)](#), [show ag](#), [show ag nport-utilization](#)

# clear arp

Clears some or all Address Resolution Protocol (ARP) entries.

## Syntax

```
clear arp [ no-refresh ] [ vrf vrf-name ] [ rbridge-id rbridge-id ]
```

```
clear arp <N>gigabitethernet rbridge-id / slot / port [ no-refresh ] [ vrf vrf-name ]
```

```
clear arp { ip ip-address | ve vlan-id } [ no-refresh ] [ vrf vrf-name ] [ rbridge-id rbridge-id ]
```

```
clear arp port-channel index [ no-refresh ]
```

```
clear arp slot number [ vrf vrf-name ] [ ip-address ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**ip** *ip-address*

Specifies a next-hop IP address.

**ve** *vlan-id*

Specifies a virtual Ethernet (VE) interface.

**no-refresh**

Clears the ARP cache without resending ARP requests to the local hosts.

**vrf** *vrf-name*

Specifies a VRF instance.

**port-channel** *index*

Specifies a port-channel interface.

**slot** *number*

Specifies a linecard slot number.

*ip-address*

Specifies a next-hop IP address.

**rbridge-id** *rbridge-id*

Specifies an RBridge.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the **no-refresh** keyword is not included, ARP requests are automatically triggered for the cleared entries. To avoid this triggering, include the **no-refresh** keyword. It is required to include the **no-refresh** keyword, in case the number of ARP entries reaches the system threshold

## Examples

The following example clears all ARP entries on the device.

```
device# clear arp
```

# clear bfd counters

Clears Bidirectional Forwarding Detection (BFD) counters.

## Syntax

```
clear bfd counters ip-address [ interface { <N>gigabitethernet rbridge-id/slot/port | loopback number | tunnel number | ve
vlan_id } [ rbridge-id { rbridge-id | all } ]
```

```
clear bfd counters ipv6-address [ interface { <N>gigabitethernet rbridge-id/slot/port | loopback number | tunnel number | ve
vlan_id } [ rbridge-id { rbridge-id | all } ]
```

```
clear bfd counters interface { <N>gigabitethernet rbridge-id/slot/port | loopback number | tunnel number | ve vlan_id }
[ rbridge-id { rbridge-id | all } ]
```

```
clear bfd counters rbridge-id { rbridge-id | all }
```

## Parameters

*ip-address*

Clears BFD over IPv4 information.

**interface**

Specifies an interface.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**loopback number**

Specifies a loopback port number in the range of 1 to 255.

**tunnel number**

Specifies a tunnel. The number of available channels range from 1 through 6144.

**ve vlan\_id**

Specifies a virtual Ethernet (VE) interface. (Refer to the Usage Guidelines.)

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

*ipv6-address*

Clears BFD over IPv6 information.

## Modes

Privileged EXEC mode

## Examples

This example clears the IPv4 BFD counters on a VE interface.

```
device# clear bfd counters 10.1.1.1 interface ve 22
```

## History

Release version	Command history
6.0.1	This command was introduced.

# clear bgp evpn l2routes

Clears BGP EVPN routes from the MAC-VRF table and triggers best route calculation.

## Syntax

**clear bgp evpn l2routes**

**clear bgp evpn l2routes type arp** *ip address mac mac address ethernet-tag tag-id*

**clear bgp evpn l2routes type inclusive-multicast** [ **ethernet-tag tag-id ip address rbridge-id** { *rbridge\_id* | **all** } ] [**rbridge-id** { *rbridge\_id* | **all** } ]

**clear bgp evpn l2routes type mac** *mac address ethernet-tag tag-id*

**clear bgp evpn l2routes type nd** *IPv6 address mac mac address ethernet-tag tag-id*

## Parameters

### arp

Specifies address-resolution protocol (ARP) routes.

### *ip address*

Specifies the IP address.

### mac *mac address*

Specifies Media Access Control (MAC) routes and a MAC address. The valid format is HHHH.HHHH.HHHH.

### ethernet-tag *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

### inclusive-multicast

Specifies inclusive multicast routes.

### rbridge-id

Specifies one or all RBridges.

### *rbridge\_id*

Specifies an RBridge.

### all

Specifies all RBridges.

### mac

Specifies MAC routes.

### nd

Specifies neighbor-discovery (ND) routes.

### *ipv6 address*

Specifies the IPv6 address.

## Modes

Privileged EXEC mode



## Examples

This example clears all BGP EVPN routes from the MAC-VRF table.

```
device# clear bgp evpn l2routes
```

This example clears all BGP EVPN ARP routes from the MAC-VRF table.

```
device# clear bgp evpn l2routes type arp
```

This example clears a specified BGP EVPN ARP route from the MAC-VRF table.

```
device# clear bgp evpn l2routes type arp 10.0.0.6 mac 0000.abba.baba ethernet-tag 0
```

This example clears all BGP EVPN MAC routes from the MAC-VRF table.

```
device# clear bgp evpn l2routes type mac
```

This example clears a specified BGP EVPN MAC route from the MAC-VRF table.

```
device# clear bgp evpn l2routes type mac 0000.abba.baba ethernet-tag 0
```

This example clears all BGP EVPN ND routes from the MAC-VRF table.

```
device# clear bgp evpn l2routes type nd
```

This example clears a specified BGP EVPN ND route from the MAC-VRF table.

```
device# clear bgp evpn l2routes type nd 2000::1 mac 0000.abba.baba ethernet-tag 0
```

This example clears all BGP EVPN inclusive multicast routes from the MAC-VRF table.

```
device# clear bgp evpn l2routes type inclusive-multicast
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.0.1	The <b>inclusive-multicast</b> keyword was added.

# clear bgp evpn local routes

Clears local routes from the BGP EVPN route table and triggers best route calculation for the specified routes.

## Syntax

`clear bgp evpn local routes`

`clear bgp evpn local routes type { arp | ipv4-prefix | ipv6-prefix | mac | nd }`

## Parameters

**arp**

Specifies address-resolution protocol (ARP) routes.

**ipv4-prefix**

Specifies IPv4 prefix routes.

**ipv6-prefix**

Specifies IPv6 prefix routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor-discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

This example clears all local routes from the BGP EVPN route table.

```
device# clear bgp evpn local routes
```

This example clears local ARP routes from the BGP EVPN route table.

```
device# clear bgp evpn local routes type arp
```

This example clears local MAC routes from the BGP EVPN route table.

```
device# clear bgp evpn local routes type mac
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear bgp evpn mac-route dampening

Clears information about dampened Media Access Control (MAC) routes for BGP EVPN.

## Syntax

```
clear bgp evpn mac-route dampening all
```

```
clear bgp evpn mac-route dampening mac-address mac address { ipv4-address | ipv6-address } ethernet-tag tag-id
```

## Parameters

**all**

Specifies all dampened routes.

**mac-address** *mac address*

Specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

*ipv4-address*

Specifies an IPv4 address.

*ipv6-address*

Specifies an IPv6 address.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

## Modes

Privileged EXEC mode

## Examples

This example clears information about all dampened MAC routes.

```
device# clear bgp evpn mac-route dampening all
```

This example clears information about a specified dampened MAC route.

```
device# clear bgp evpn mac-route dampening mac-address 0000.abba.baba 10.0.0.0 ethernet-tag 0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear bgp evpn neighbor

Requests a dynamic refresh of BGP EVPN connections or routes from a neighbor, with a variety of options.

## Syntax

```
clear bgp evpn neighbor { all | ipv4-addr | ipv6-addr } soft [ in | out ]
```

```
clear bgp evpn neighbor { all | ipv4-addr | ipv6-addr } soft-outbound
```

## Parameters

### all

Resets and clears all BGP EVPN connections to all neighbors.

### ipv4-addr

Specifies an IPv4 address.

### ipv6-addr

Specifies an IPv6 address.

### soft

Refreshes routes received from or sent to the neighbor.

#### in

Refreshes received routes.

#### out

Refreshes sent routes.

### soft-outbound

Refreshes all outbound routes by applying new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor.

### NOTE

Use **soft-outbound** only if the outbound policy is changed. This operand updates all outbound routes by applying the new or changed filters. However, the device sends to the neighbor only the existing routes that are affected by the new or changed filters. The **soft out** operand updates all outbound routes and then sends the entire route table on the device to the neighbor after the device changes or excludes the routes affected by the filters.

## Modes

Privileged EXEC mode

## Examples

This example refreshes all BGP EVPN neighbor connections.

```
device# clear bgp evpn neighbor all
```

This example clears BGP EVPN connections with a specified IPv6 address.

```
device# clear bgp evpn neighbor 2001::1
```

This example refreshes routes received from a neighbor with the IP address 10.0.0.1.

```
device# clear bgp evpn neighbor 10.0.0.1 soft in
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear bgp evpn routes

Clears routes from the BGP EVPN route table and resets the routes.

## Syntax

**clear bgp evpn routes**

**clear bgp evpn routes type arp** *ip address* **mac** *mac address* **ethernet-tag** *tag-id*

**clear bgp evpn routes type ipv4-prefix** *ip address/mask*

**clear bgp evpn routes type ipv6-prefix** *ipv6 address/mask*

**clear bgp evpn routes type mac** *mac address* **ethernet-tag** *tag-id*

**clear bgp evpn routes type nd** *IPv6 address* **mac** *mac address* **ethernet-tag** *tag-id*

## Parameters

**arp**

Specifies address-resolution protocol (ARP) routes.

*ip address*

Specifies an IP address.

**mac** *mac address*

Specifies Media Access Control (MAC) routes and a MAC address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**ipv4-prefix**

Specifies IPv4 prefix routes.

*IPv4 address/mask*

Specifies an IPv4 address and mask.

**ipv6-prefix**

Specifies IPv6 prefix routes.

*IPv6 address/mask*

Specifies an IPv6 address and mask.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor-discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

This example clears all routes from the BGP EVPN route table.

```
device# clear bgp evpn routes
```

This example clears all ARP routes from the BGP EVPN route table.

```
device# clear bgp evpn routes type arp
```

This example clears a specified MAC route from the BGP EVPN route table.

```
device# clear bgp evpn routes type mac 000.abba.baba ethernet-tag 0
```

## History

Release version	Command history
7.0.0	This command was introduced.

## clear counters access-list

For a given network protocol and inbound/outbound direction, clears ACL statistical information. You can clear all statistics for a specified ACL or only for that ACL on a specified interface. You can also clear statistical information for all ACLs bound to a specified switch interface, VLAN, VE, or VXLAN overlay gateway. You can also clear statistical information for receive-path ACLs on a specified RBridge or on all RBridges.

### Syntax

```
clear counters access-list interface { <N>gigabitethernet rbridge_id / slot / port | port-channel index | vlan vlan_id } { in | out }
clear counters access-list interface ve ve_id { in | out } [ rbridge-id { rbridge_id | all } ]
clear counters access-list { ip | ipv6 } [ acl-name { in | out } ]
clear counters access-list { ip | ipv6 } acl-name interface { <N>gigabitethernet rbridge_id / slot / port | port-channel index | ve ve_id } { in | out }
clear counters access-list { ip | ipv6 } acl-name rbridge-id { rbridge_id | all } in
clear counters access-list rbridge-id { rbridge_id | all } in
clear counters access-list mac [ acl-name { in | out } ]
clear counters access-list mac acl-name interface { <N> gigabitethernet rbridge_id / slot / port | port-channel index | vlan vlan_id } { in | out }
clear counters access-list overlay-gateway overlay_gateway_name in
```

### Parameters

#### interface

Filter by interface.

#### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge\_id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

#### port-channel *number*

Specifies a port-channel. Available channels range from 1 through 6144.

#### vlan *vlan\_id*

(Available only on Layer 2) Specifies a VLAN.

#### ve *ve\_id*

(Available only on Layer 3) Specifies a virtual Ethernet (VE) interface.



**ip | ipv6 | mac**

Specifies the network protocol.

*acl-name*Specifies the ACL name. To clear statistics on all counters of an ACL-type, do not specify *acl-name*.**rbridge-id**

Specifies one or all RBridges.

*rbridge\_id*

Specifies an RBridge.

**all**

Specifies all RBridges.

**overlay-gateway** *overlay\_gateway\_name*

Specifies a VXLAN overlay-gateway.

**in | out**

Specifies the binding direction (incoming or outgoing).

## Modes

Privileged EXEC mode

## Examples

The following example clears ACL statistics for the 10-gigabit Ethernet interface 5/0/1.

```
device# clear counters access-list interface tengigabitethernet 5/0/1
```

The following example clears ACL statistics for the MAC ACL named test on a specified interface.

```
device# clear counters access-list mac test interface tengigabitethernet 5/0/1
```

The following example clears ACL statistics for the MAC ACL named test on all interfaces on which this ACL is applied.

```
device# clear counters access-list mac test
```

The following example clears counters for the IPv4 ACL named test on a specified interface.

```
device# clear counters access-list ip test interface tengigabitethernet 6/0/1
```

The following example clears ACL statistics for the IPv4 ACL named test on all interfaces on which it is applied.

```
device# clear counters access-list ip test
```

The following example clears incoming ACL statistics for an IPv6 ACL on RBridge 122 on a virtual Ethernet (VE) interface.

```
device# clear counters access-list ipv6 ip_acl_3 interface ve 10 in rbridge-id 122
```

The following example clears incoming statistics for a specified ACL.

```
device# clear counters access-list ipv6 ipv6-receive-acl-example in
```

The following example clears (incoming) statistics for all ACLs applied to a specified overlay gateway.

```
device# clear counters access-list overlay-gateway gw121 in
```

# clear counters all

Clears the IP counter statistics and ASIC counters on all interfaces on the switch.

## Syntax

```
clear counters all rbridge-id { rbridge_id | all }
```

## Parameters

### rbridge-id

Specifies one or all RBridges.

### *rbridge\_id*

Specifies an RBridge.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode.

## Examples

To clear all IP counter statistics and ASIC counters for all interfaces on all RBridges in a cluster:

```
device# clear counters all rbridge-id all
```

## History

Release version	Command history
7.1.0	This command was modified to include support for clearing ASIC counters.

# clear counters interface

Clears the IP counter statistics on a specified interface on the switch.

## Syntax

```
clear counters interface { fcoe { vn-number/rbridge-id/port | all } | fibrechannel { rbridge-id/slot/port } | port-channel number |
[ <N>gigabitethernet { rbridge-id/slot/port | all } ] | vlan { vlan_id } }
```

## Parameters

### interface

Specifies the use of the *fcoe*, *port-channel*, *fibrechannel*, *fortygigabitethernet*, *gigabitethernet*, *tengigabitethernet*, or *vlan* keyword.

**fcoe***vn-number/rbridge-id/port*

*vn-number/rbridge-id/port* Specifies the FCOE interface name.

*vn-number*

Specifies the VN number for FCoE.

*rbridge-id*

Specifies the RBridge ID.

*port*

Specifies a valid port number.

**all**

Clears counters for all FCoE interfaces.

**fibrechannel***rbridge-id/slot/port*

*rbridge-id/slot/port* Specifies a valid Fibre Channel interface.

*rbridge-id*

Specifies the RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

clear counters interface

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies the interface is a port-channel. The number of available channels range from 1 through 6144.

**vlan** *vlan\_id*

Specifies a VLAN interface. (Refer to the Usage Guidelines.)

## Modes

Privileged EXEC mode

## Usage Guidelines

The **clear counters all** command does not clear counters for any of the protocol daemon stats such as LLDP, LACP, MSTP, and so on.

For Extreme VDX switches, the slot number is always 0 (zero).

# clear counters slot-id

Clears the IP counter statistics on a specified slot in the chassis.

## Syntax

```
clear counters slot-id num
```

## Parameters

*num*

Specifies a valid integer.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **clear counters all** command does not clear counters for any of the protocol daemon statistics such as LLDP, LACP, MSTP, and so on.

For Extreme VDX switches, the slot number is always 0 (zero).

# clear counters storm-control

Clears all broadcast, unknown unicast, and multicast (BUM) related counters in the system.

## Syntax

```
clear counters storm-control
```

```
clear counters storm-control [ broadcast | multicast | unknown-unicast ] [ interface <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### broadcast

Clears all BUM-related counters in the system for the broadcast traffic type.

### multicast

Clears all BUM-related counters in the system for the multicast traffic type.

### unknown-unicast

Clears all BUM-related counters in the system for the unknown-unicast traffic type.

**interface** <N>**gigabitethernet** *rbridge-id/slot/port*

Specifies an interface type, followed by the RBridge ID/slot/port, for which to clear all BUM-related counters in the system for the specified traffic type. Use this parameter to clear counters on a per-port basis.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *N* **gigabitethernet** <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

## Modes

Privileged EXEC mode.

## Usage Guidelines

This command clears the counters for broadcast, unknown-unicast, and multicast traffic for the entire system, for specified traffic types, for specified interfaces, or for specified traffic types on specified interfaces.

## Examples

Clear counters for broadcast traffic on the 10-gigabit Ethernet interface.

```
device# clear counters storm-control broadcast interface tengigabitethernet 102/4/1
```

Clear counters for all traffic types enabled on the 10-gigabit Ethernet interface.

```
device# clear counters storm-control interface tengigabitethernet 102/4/1
```

Clear counters for all multicast traffic in the system.

```
device# clear counters storm-control multicast
```

Clear all BUM-related counters in the system.

```
device# clear counters storm-control
```

# clear dot1x statistics

Clears all accumulated dot1x port authentication statistics on all ports.

## Syntax

```
clear dot1x statistics
```

## Modes

Privileged EXEC mode

## Examples

To clear dot1x statistics:

```
device# clear dot1x statistics
```

## Related Commands

[clear dot1x statistics interface](#)



# clear dot1x statistics interface

Clears all dot1x statistics for a specified interface port.

## Syntax

```
clear dot1x statistics interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **Ngigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

To clear dot1x statistics on a port:

```
device# clear dot1x statistics interface tengigabitethernet 0/16
```

## Related Commands

[clear dot1x statistics](#)

# clear edge-loop-detection

Re-enables all ports disabled by ELD and clears all ELD statistics.

## Syntax

```
clear edge-loop-detection [ rbridge-id rbridge-id ]
```

```
clear edge-loop-detection interface { <N>gigabitethernet { rbridge-id/slot/port } | port-channel num }
```

## Parameters

*rbridge-id*

A unique identifier for the switch. Values are from 1 through 239.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *N***gigabitethernet** with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *num*

Specifies the interface is a port-channel. The number of available channels range from 1 through 6144.

## Modes

ELD configuration mode

## Usage Guidelines

This operation is typically performed after correcting a configuration error that caused ELD to disable ports.

This functionality detects Layer 2 loops only.

If the *rbridge-id* is specified, it clears edge-loop-detection from the specific node. Otherwise, it clears edge-loop-detection from all nodes in the VCS cluster.

## Related Commands

[protocol edge-loop-detection](#), [show edge-loop-detection rbridge-id](#), [show edge-loop-detection interface](#)

# clear ip arp inspection statistics

Clears dynamic ARP inspection (DAI) statistics for all DAI-enabled VLANs.

## Syntax

```
clear ip arp inspection statistics
```

## Modes

Privileged EXEC mode

## Usage Guidelines

The capacity of each statistic counter is 64 bits, beyond which such a counter is reset to zero.

## Examples

The following example clears DAI statistics for all DAI-enabled VLANs.

```
device# clear ip arp inspection statistics
```

## History

Release version	Command history
6.0.1	This command was introduced.

# clear ip arp suppression-cache

Clears the IPv4 ARP-suppression cache and downloads the current forwarding database from BGP-EVPN. You can also clear the cache for a specified VLAN.

## Syntax

```
clear ip arp suppression-cache [ vlan vlan-id ]
```

## Parameters

**vlan** *vlan-id*  
Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

Running this command might impact traffic.

## Examples

The following example clears the ARP-suppression cache.

```
device# clear ip arp suppression-cache
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear ip arp suppression-statistics

Clears ARP-suppression statistical information. You can also clear statistics for a specified VLAN.

## Syntax

```
clear ip arp suppression-statistics [ vlan vlan-id ]
```

## Parameters

**vlan** *vlan-id*

Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## Examples

The following example clears all ARP-suppression statistics.

```
device# clear ip arp suppression-statistics
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear ip bgp dampening

Reactivates suppressed BGP4 routes.

## Syntax

```
clear ip bgp dampening [ ip-addr { / mask } ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*ip-addr*

IPv4 address of a specified route in dotted-decimal notation.

*mask*

(Optional) IPv4 mask of a specified route in CIDR notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example unsuppresses all suppressed BGP4 routes.

```
device# clear ip bgp dampening
```

This example unsuppresses suppressed BGP4 routes for VRF "red".

```
device# clear ip bgp dampening vrf red
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ip bgp flap-statistics

Clears the dampening statistics for a BGP4 route without changing the dampening status of the route.

## Syntax

```
clear ip bgp flap-statistics [ ip-addr { / mask } | neighbor ip-addr | regular-expression string ] [ rbridge-id { rbridge-id | all } ]
[ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*ip-addr*

IPv4 address of a specified route in dotted-decimal notation.

*mask*

(Optional) IPv4 mask of a specified route in CIDR notation.

**neighbor**

Clears dampening statistics only for routes learned from the specified neighbor.

*ip-addr*

IPv4 address of the neighbor.

**regular-expression**

Specifies a regular expression.

*string*

Regular expression.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example clears the dampening statistics for a BGP4 route.

```
device# clear ip bgp flap-statistics 10.0.0.0/16
```

This example clears the dampening statistics for a BGP4 route for VRF "red".

```
device# clear ip bgp flap-statistics 10.0.0.0/16 vrf red
```

clear ip bgp flap-statistics

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.



# clear ip bgp local routes

Clears BGP4 local routes from the IP route table and resets the routes.

## Syntax

```
clear ip bgp local routes [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

### vrf *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example clears all BGP4 local routes.

```
device# clear ip bgp local routes
```

This example clears BGP4 local routes for VRF "red".

```
device# clear ip bgp local routes vrf red
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ip bgp neighbor

Requests a dynamic refresh of BGP4 connections or routes from a neighbor, with a variety of options.

## Syntax

```
clear ip bgp neighbor [ all | as-num | peer-group-name | ip-addr ] [ last-packet-with-error | notification-errors | soft [ in
  [ prefix-filter ] | out ] | soft-outbound | traffic ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

### all

Resets and clears all BGP4 connections to all neighbors.

### as-num

Clears all BGP4 connections within this autonomous system. Range is from 1 through 4294967295.

### peer-group-name

Clears all BGP4 connections in this peer group. Range is from 1 through 63 characters.

### ip-addr

Clears all BGP4 connections with this IPv4 address, in dotted-decimal notation.

### last-packet-with-error

Clears all BGP4 connections identified as having the last packet received with an error.

### notification-errors

Clears all BGP4 connections identified as having notification errors.

### soft

Refreshes routes received from or sent to the neighbor.

#### in

Refreshes received routes.

#### prefix-filter

Refreshes Outbound Route Filters (ORFs) that are prefix-based.

#### out

Refreshes sent routes.

### soft-outbound

Refreshes all outbound routes by applying new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor.

#### NOTE

Use **soft-outbound** only if the outbound policy is changed. This operand updates all outbound routes by applying the new or changed filters. However, the device sends to the neighbor only the existing routes that are affected by the new or changed filters. The **soft out** operand updates all outbound routes and then sends the entire BGP4 route table on the device to the neighbor after the device changes or excludes the routes affected by the filters.

### traffic

Clears the counters (resets them to 0) for BGP4 messages.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf *vrf-name***

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example refreshes all BGP4 neighbor connections.

```
device# clear ip bgp neighbor all
```

This example refreshes all BGP4 neighbor connections for VRF "red".

```
device# clear ip bgp neighbor all vrf red
```

## History

Release version	Command history
6.0.1	The <b>vrf <i>vrf-name</i></b> parameter was added to support Multi-VRF.

# clear ip bgp routes

Clears BGP4 routes from the IP route table and resets the routes.

## Syntax

```
clear ip bgp routes [ip-addr [/ mask]][ rbridge-id { rbridge-id | all }][ vrf vrfname [ rbridge-id { rbridge-id | all } ]]
```

## Parameters

*ip-addr*

IPv4 address of a specified route in dotted-decimal notation.

*mask*

(Optional) IPv4 mask of a specified route in CIDR notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of the VRF instance to associate with subsequent address-family configuration mode commands.

## Modes

Privileged EXEC mode

## Examples

This example clears all BGP4 routes.

```
device# clear ip bgp routes 10.0.0.0/16
```

This example clears BGP4 routes for VRF instance "red":

```
device# clear ip bgp routes 10.0.0.0/16 vrf red
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ip bgp traffic

Clears the BGP4 message counter for all neighbors.

## Syntax

```
clear ip bgp traffic [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

### **rbridge-id**

Specifies one or all RBridges.

*rbridge\_id*

Specifies an RBridge.

**all**

Specifies all RBridges.

### **vrf** *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

To clear the BGP4 message counters:

```
device# clear ip bgp traffic
```

# clear ip dhcp relay statistics

Clears IP DHCP Relay statistics.

## Syntax

**clear ip dhcp relay statistics**

**clear ip dhcp relay statistics** [**ip-address** *ip-address*] **rbridge-id** { *rbridge-id* | *range* | **all** }

## Command Default

DHCP relay statistics are present on the DHCP server.

## Parameters

**ip-address** *ip-address*

IPv4 address of DHCP server where client requests are to be forwarded.

**rbridge-id**

Specifies an RBridge, multiple RBridges, or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

*range*

Specifies multiple RBridge IDs. You can specify a range (for example, 3-5), a comma-separated list (for example, 1,3,5,6), or you can combine a range with a list (for example, 1-5,6,8). In a range string, no spaces are allowed.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to clear IP DHCP Relay statistics for a specific IP DHCP Relay address or all addresses on a local switch, specific switches, or all nodes in a cluster.

If the **rbridge-id** parameter is omitted, statistics are cleared for the local switch. If the **ip\_address** parameter is omitted, statistics are cleared for all configured addresses on defined switches.

## Examples

The following example clears statistics for IP DHCP Relay addresses on RBridge IDs 1, 3, and 5.

```
device# clear ip dhcp relay statistics rbridge-id 1,3,5
```

The following example clears statistics for IP DHCP Relay address 10.1.0.1 configured on RBridge IDs 1, 3, and 5.

```
device# clear ip dhcp relay statistics ip-address 10.1.0.1 rbridge-id 1,3,5
```

## Related Commands

[show ip dhcp relay statistics](#)

# clear ip fabric-virtual-gateway

Clears IP Fabric-Virtual-Gateway protocol statistics globally or for a Virtual Ethernet (VE) interface.

## Syntax

```
clear ip fabric-virtual-gateway { all | interface ve vlan-id }
```

## Command Default

None

## Parameters

**all**

Retriggers the election of ARP responders for all sessions.

**interface ve *vlan-id***

Clears IP Fabric-Virtual-Gateway configurations for the specified VE interface. The range is from 1 through 8191.

## Modes

Privileged EXEC mode

## Usage Guidelines

A **clear** command must be issued to retrigger the election of a new ARP responder.

## Examples

The following example clears the IP Fabric-Virtual-Gateway protocol globally.

```
device# clear ip fabric-virtual-gateway all
```

The following example clears the IP Fabric-Virtual-Gateway protocol for a specific VE interface.

```
device# clear ip fabric-virtual-gateway interface ve 2000
```

## History

Release version	Command history
5.0.1	This command was introduced.



# clear ip igmp groups

Clears information related to learned groups in the IGMP module.

## Syntax

```
clear ip igmp groups [ A.B.C.D ] [ interface { port-channel number | vlan vlan_id } | <N>gigabitethernet rbridge-id/slot/port |
port-channel number | vlan vlan_id | ve vlan_id ] [ rbridge rbridge-id ]
```

## Parameters

*A.B.C.D*

Specifies the group address, as a subnet number in dotted decimal format (for example, 10.0.0.1), as the allowable range of addresses included in the multicast group.

**interface**

Specifies an interface.

<N> **gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel. The number of available channels range from 1 through 6144.

**vlan** *vlan\_id*

Specifies a VLAN. Refer to "Usage Guidelines" below.

**ve** *vlan\_id*

Specifies groups on the specified virtual Ethernet (VE) interface. (Refer to the Usage Guidelines.)

**rbridge** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

```
clear ip igmp groups
```

## Examples

To clear information for all groups in the IGMP protocol:

```
device# clear ip igmp groups
```

# clear ip igmp statistics interface

Clears statistical information related to the IGMP database.

## Syntax

```
clear ip igmp statistics interface { <N>gigabitethernet rbridge-id/slot/port | port-channel number | ve vlan_id | vlan vlan_id
[ rbridge-id { rbridge-id | all } ] | rbridge-id { rbridge-id | all } }
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel *number*

Specifies the interface is a port-channel. The number of available channels range from 1 through 6144.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface. (Refer to the Usage Guidelines.)

### vlan *vlan\_id*

Specifies a VLAN interface. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled. (Refer to the Usage Guidelines.)

### rbridge-id

Specifies one or all RBridges.

*rbridge\_id*

Specifies an RBridge.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command can be used with user-configurable VLAN IDs.

When the **rbridge-id** option is specified, details for the VE interface on that particular RBridge are cleared. If **rbridge-id** is not specified, details for the VE interface on the node on which the command is executed are cleared. When **rbridge-id all** is specified, all ve interfaces with that **rbridge-id** from all the nodes in the cluster are cleared.

## Examples

The following example clears statistics information for a VLAN in the IGMP protocol.

```
device# clear ip igmp statistics interface vlan 11
```

## History

Release version	Command history
7.1.0	This command was modified to support the <b>all</b> keyword for <b>rbridge-id</b> for all interfaces.

# clear ip ospf

Clears OSPF process, counters, neighbors, or routes.

## Syntax

```
clear ip ospf all [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
clear ip ospf counters <N>gigabitethernet [ rbridge-id / ] slot / port [ vrf vrf-name ]
```

```
clear ip ospf counters { loopback number | port-channel number | ve vlan_id } [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
clear ip ospf neighbor { A.B.C.D | all }
```

```
clear ip ospf routes { A.B.C.D | all }
```

## Parameters

### all

Clears all counters.

### counters

Clears all counters or clears the counters of an interface that you specify.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

(Optional) Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### loopback *number*

Specifies a loopback port number. The range is from 1 through 255.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

clear ip ospf

**vrf name**

Clears the specified VRF.

**neighbor**

Clears the specified neighbor, or clears all neighbors.

*A.B.C.D*

Specifies the IP address of the neighbor to clear.

**all**

Clears all neighbors.

**routes**

Clears matching routes or clears all routes.

*A.B.C.D*

Clears all routes that match the prefix and mask that you specify.

**all**

Clears all routes.

## Modes

Privileged EXEC mode

## Examples

The following example restarts the OSPF processes.

```
device# clear ip ospf all
```

# clear ip pim mcache

Clears the Protocol Independent Multicast forwarding cache.

## Syntax

```
clear ip pim mcache [ IP-addr [ IP-addr ] ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*IP-addr*

Group or source IPv4 address. One or two IP addresses (unicast or multicast) can be specified.

**rbridge-id**

Filter by RBridge ID.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

If you do not include the **rbridge-id** keyword, this command clears output for the current node only.

## Examples

The following example clears the multicast cache summary for the current node.

```
device# clear ip pim mcache
```

The following example clears the multicast cache summary for all RBridges in the cluster.

```
device# clear ip pim mcache rbridge-id all
```

## History

Release version	Command history
6.0.1a	The <b>rbridge-id</b> options were added.

## Related Commands

[clear ip pim rp-map](#), [clear ip pim traffic](#)

# clear ip pim rp-map

Clears the static multicast forwarding table.

## Syntax

```
clear ip pim rp-map [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Filter by RBridge ID.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command should be used after the static Rendezvous Point configuration has been changed. This allows Protocol Independent Multicast to immediately start using the new Rendezvous Point, rather than waiting for the old information to expire.

If you do not include the **rbridge-id** keyword, this command clears output for the current node only.

## Examples

The following example clears the RP-to-group mappings for the current node.

```
device# clear ip pim rp-map
```

The following example clears the RP-to-group mappings, for all RBridges in the cluster.

```
device# clear ip pim rp-map rbridge-id all
```

## History

Release version	Command history
6.0.1a	The <b>rbridge-id</b> options were added.

## Related Commands

[clear ip pim mcache](#), [clear ip pim traffic](#)



# clear ip pim traffic

Clears the Protocol Independent Multicast (PIM) traffic counters.

## Syntax

```
clear ip pim traffic [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Filter by RBridge ID.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

If you do not include the **rbridge-id** keyword, this command clears output for the current node only.

## Examples

The following example clears PIM traffic statistics for the current node.

```
device# clear ip pim rp-map
```

The following example clears PIM traffic statistics, for all RBridges in the cluster.

```
device# clear ip pim rp-map rbridge-id all
```

## History

Release version	Command history
6.0.1a	The <b>rbridge-id</b> options were added.

## Related Commands

[clear ip pim rp-map](#), [clear ip pim mcache](#)

# clear ip route

Clears a specified route or all IP routes in the IP routing tables.

## Syntax

```
clear ip route A.B.C.D/L [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
```

```
clear ip route all [ import src-vrf-name ] [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
```

```
clear ip route slot line_card_number [ A.B.C.D/L ] [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*A.B.C.D/L*

Specifies an IPv4 address and prefix length.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges (in context).

**vrf***vrf-name*

Specifies a VRF instance from which the user is currently retrieving routes.

**all**

Specifies all routes (when used as the first parameter following the **clear ip route** command).

**import**

Specifies imported IPv4 routes.

*src-vrf-name*

Specifies a VRF instance from which routes are leaked.

**slot** *line\_card\_number*

Specifies a line card.

## Modes

Privileged EXEC mode

## Usage Guidelines

A standard mask is required (for example, 10.1.0.0/16, 10.1.1.0/24, 10.0.0.0/32).

## Examples

To clear the IP route specified by the prefix 192.158.1.1/24:

```
device# clear ip route 192.158.1.1/24
```

## History

Release version	Command history
6.0.0	This command was modified to support the <b>vrf</b> keyword.
6.0.1a	The source VRF name was specified as optional.

# clear ipv6 bgp dampening

Reactivates all suppressed BGP4+ routes.

## Syntax

```
clear ipv6 bgp dampening [ ipv6-addr { / mask } ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*ipv6-addr*

IPv6 address of a specified route in dotted-decimal notation.

*mask*

(Optional) IPv6 mask of a specified route in CIDR notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example unsuppresses all suppressed BGP4+ routes.

```
device# clear ipv6 bgp dampening
```

This example unsuppresses suppressed BGP4+ routes for VRF "red".

```
device# clear ipv6 bgp dampening vrf red
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ipv6 bgp flap-statistics

Clears route-flap statistics for BGP4+ routes.

## Syntax

```
clear ipv6 bgp flap-statistics [ ipv6-addr { / mask } | neighbor ipv6-addr | regular-expression string ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*ipv6-addr*

IPv6 address of a specified route in dotted-decimal notation.

*mask*

(Optional) IPv6 mask of a specified route in CIDR notation.

**neighbor**

Clears route-flap statistics only for routes learned from the specified neighbor.

*ipv6-addr*

IPv6 address of the neighbor.

**regular-expression**

Specifies a regular expression.

*string*

Regular expression.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example clears all dampening statistics for a BGP4+ route.

```
device# clear ipv6 bgp flap-statistics
```

This example clears the dampening statistics for a BGP4+ route for VRF "red".

```
device# clear ipv6 bgp flap-statistics vrf red
```

clear ipv6 bgp flap-statistics

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ipv6 bgp local routes

Clears BGP4+ local routes from the IP route table and resets the routes.

## Syntax

```
clear ipv6 bgp local routes [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

#### **all**

Specifies all RBridges.

### vrf *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example clears all BGP4+ local routes.

```
device# clear ipv6 bgp local routes
```

This example clears BGP4+ local routes for VRF "red".

```
device# clear ipv6 bgp local routes vrf red
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ipv6 bgp neighbor

Requests a dynamic refresh of BGP4+ connections or routes from a neighbor, with a variety of options.

## Syntax

```
clear ipv6 bgp neighbor [ all | as-num | peer-group-name | ipv6-addr ] [ last-packet-with-error | notification-errors | soft [ in
  [ prefix-filter ] | out ] | soft-outbound | traffic ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

### all

Resets and clears all BGP4+ connections to all neighbors.

### as-num

Clears all BGP4+ connections within this autonomous system. Range is from 1 through 4294967295.

### peer-group-name

Clears all BGP4+ connections in this peer group. Range is from 1 through 63 characters.

### ipv6-addr

Clears all BGP4+ connections with this IPv6 address, in dotted-decimal notation.

### last-packet-with-error

Clears all BGP4+ connections identified as having the last packet received with an error.

### notification-errors

Clears all BGP4+ connections identified as having notification errors.

### soft

Refreshes routes received from or sent to the neighbor.

#### in

Refreshes received routes.

#### prefix-filter

Refreshes Outbound Route Filters (ORFs) that are prefix-based.

#### out

Refreshes sent routes.

### soft-outbound

Refreshes all outbound routes by applying new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor.

#### NOTE

Use **soft-outbound** only if the outbound policy is changed. This operand updates all outbound routes by applying the new or changed filters. However, the device sends to the neighbor only the existing routes that are affected by the new or changed filters. The **soft out** operand updates all outbound routes and then sends the entire BGP4+ route table on the device to the neighbor after the device changes or excludes the routes affected by the filters.

### traffic

Clears the counters (resets them to 0) for BGP4+ messages.



**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf vrf-name**

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example refreshes all BGP4+ neighbor connections.

```
device# clear ipv6 bgp neighbor all
```

This example resets all the counters for BGP4+ messages.

```
device# clear ipv6 bgp neighbor all traffic
```

This example clears BGP4+ connections with a specified IPv6 address.

```
device# clear ipv6 bgp neighbor 2001::1
```

This example clears BGP4+ connections with a specified peer group.

```
device# clear ipv6 bgp neighbor P1
```

This example clears BGP4+ connections with a specified peer group for VRF "red".

```
device# clear ipv6 bgp neighbor P1 vrf red
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# clear ipv6 bgp routes

Clears BGP4+ routes from the IP route table and resets the routes.

## Syntax

```
clear ipv6 bgp routes [ ipv6-addr [ / mask ] ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*ipv6-addr*

IPv6 address of a specified route in dotted-decimal notation.

*mask*

IPv6 mask of a specified route in CIDR notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example clears specific BGP4+ routes.

```
device# clear ipv6 bgp routes 2000::/64
```

This example clears specific BGP4+ routes for VRF "red".

```
device# clear ipv6 bgp routes 2000::/64 vrf red
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ipv6 bgp traffic

Clears the BGP4+ message counter for all neighbors.

## Syntax

```
clear ipv6 bgp traffic [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Modes

Privileged EXEC mode

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

### vrf *vrf-name*

Specifies the name of a VRF instance.

## Examples

This example clears all BGP4+ message counters.

```
device# clear ipv6 bgp traffic
```

This example clears BGP4+ message counters for VRF "red".

```
device# clear ipv6 bgp traffic vrf red
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# clear ipv6 counters

Clears IPv6 counters on on all interfaces or on a specified interface.

## Syntax

```
clear ipv6 counters [ all | interface { <N>gigabitethernet rbridge-id/slot/port | loopback port_number | ve vlan_id [ rbridge-id
[ all | rbridge-id ] } ]
```

## Parameters

### all

Specifies all interfaces.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### rbridge-id

Specifies an RBridge ID.

#### slot

Specifies a valid slot number.

#### port

Specifies a valid port number.

### loopback

Specifies a loopback interface.

#### port\_number

Port number of the loopback interface. The range is from 1 through 255.

### ve

Specifies a virtual Ethernet (VE) interface.

#### vlan\_id

VLAN ID of the VE interface.

### rbridge-id rbridge-id

Specifies an RBridge ID.

#### all

Specifies all RBridges in the cluster.

## Modes

Privileged EXEC mode

# clear ipv6 dhcp relay statistics

Clears IPv6 DHCP Relay statistics

## Syntax

```
clear ipv6 dhcp relay statistics [ ipv6-address ipv6-address ] [ rbridge-id { rbridge-id | all } ] range ]
```

## Command Default

If the **rbridge-id** parameter is omitted, statistics clear for the local switch. If the **ip\_address** parameter is omitted, statistics clear for all configured addresses on defined switches.

## Parameters

**ip-address** *ip-addr*

IPv6 address of DHCP server where client requests are to be forwarded.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**range**

A range of RBridge IDs separated by a dash or commas, for example:

1-3 - RBridge ID 1 through 3  
 1-3, 5 - RBridge ID 1 through 3 and RBridge ID 5  
 1, 3, 5, 6 - RBridge ID 1, 3, 5, and 6

## Modes

Privileged EXEC mode

## Usage Guidelines

Clears IPv6 DHCP Relay statistics for a specific IP DHCP Relay address or all addresses on a local switch, specific switches, or all nodes in a cluster.

No spaces are allowed in the range string. The range does not need to be contiguous (for example, 1-2,5). You can also specify **all** for all RBridge IDs in a cluster.

## Examples

Clear statistics for IPv6 DHCP Relay addresses on RBridge IDs 1, 3, and 5.

```
device# clear ipv6 dhcp relay statistics rbridge-id 1,3,5
```

# clear ipv6 fabric-virtual-gateway

Clears IPv6 Fabric-Virtual-Gateway protocol statistics globally or for a Virtual Ethernet (VE) interface.

## Syntax

```
clear ipv6 fabric-virtual-gateway { all | interface ve vlan-id }
```

## Command Default

None

## Parameters

**all** Specifies all statistics.

**interface ve *vlan-id*** Clears IPv6 Fabric-Virtual-Gateway configurations for the specified VE interface. The range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

## Modes

Privileged EXEC mode

## Usage Guidelines

None

## Examples

To clear the IPv6 Fabric-Virtual-Gateway protocol statistics on VE 2000:

```
device# clear ipv6 fabric-virtual-gateway interface ve 2000
```

To clear all IPv6 Fabric-Virtual-Gateway protocol statistics:

```
device# clear ipv6 fabric-virtual-gateway all
```

## History

Release version	Command history
5.0.1	This command was introduced.

# clear ipv6 mld groups

Clears IPv6 MLDv1 group cache entries for a multicast group address or a VLAN.

## Syntax

```
clear ipv6 mld groups [ ipv6address ] [ interface vlan vlan_id ]
```

## Parameters

*ipv6address*

Specifies the IPv6 address for the group.

**interface vlan**

Specifies a VLAN ID.

*vlan\_id*

A VLAN ID. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

## Modes

Privileged EXEC mode

## Examples

To clear all IPv6 MLDv1 group cache entries:

```
device# clear ipv6 mld groups
```

To clear IPv6 MLDv1 group cache entries on a specific VLAN:

```
device# clear ipv6 mld groups interface vlan 2000
```

# clear ipv6 mld statistics

Clears IPv6 MLDv1 snooping statistics.

## Syntax

```
clear ipv6 mld statistics [ interface { <N>gigabitethernet rbridge-id/slot/port | port-channel number | ve vlan_id | vlan vlan_id
[ rbridge-id { rbridge-id | all } ] | rbridge-id { rbridge-id | all } }
```

## Parameters

### interface

Specifies an interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### port-channel *number*

Specifies the interface is a port-channel. The number of available channels range from 1 through 6144.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface. (Refer to the Usage Guidelines.)

### vlan *vlan\_id*

Specifies a VLAN interface. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled. (Refer to the Usage Guidelines.)

### rbridge-id

Specifies one or all RBridges.

#### *rbridge\_id*

Specifies an RBridge.

#### all

Specifies all RBridges.



## Modes

Privileged EXEC mode

## Examples

To clear IPv6 MLDv1 snooping statistics for a specific VLAN:

```
device# clear ipv6 mld statistics interface vlan 2000
```

## History

Release version	Command history
7.1.0	This command was modified to support the <b>all</b> keyword for <b>rbridge-id</b> for all interfaces.

# clear ipv6 nd suppression-cache

Clears the neighbor discovery (ND)-suppression cache. You can also clear the cache for a specified VLAN.

## Syntax

```
clear ipv6 nd suppression-cache [ vlan vlan-id ]
```

## Parameters

**vlan** *vlan-id*

Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## Examples

The following example clears the ND-suppression cache.

```
device# clear ipv6 nd suppression-cache
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear ipv6 nd suppression-statistics

Clears neighbor discovery (ND)-suppression statistical information. You can also clear statistics for a specified VLAN.

## Syntax

```
clear ipv6 nd suppression-statistics [ vlan vlan-id ]
```

## Parameters

**vlan** *vlan-id*

Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## Examples

The following example clears all ND-suppression statistics.

```
device# clear ipv6 nd suppression-statistics
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear ipv6 neighbor

Clears the IPv6 neighbor discovery cache on an interface.

## Syntax

```
clear ipv6 neighbor [ force-delete | no-refresh ] [ rbridge-id { rbridge-id | all } ]
clear ipv6 neighbor vrf vrf-name [ { force-delete | no-refresh } [ rbridge-id { rbridge-id | all } ] ]
clear ipv6 neighbor slot linecard-number [ ipv6-address | vrf vrf-name ]
clear ipv6 neighbor { <N>gigabitethernet rbridge-id / slot / port | port-channel number } [ vrf vrf-name ] [ force-delete | no-refresh ]
clear ipv6 neighbor ve vlan_id [ vrf vrf-name ] [ { force-delete | no-refresh } [ rbridge-id { rbridge-id | all } ] ]
clear ipv6 neighbor ipv6-address [ vrf vrf-name ] [ force-delete | no-refresh ] [ rbridge-id { rbridge-id | all } ]
clear ipv6 neighbor ipv6-address interface { <N>gigabitethernet rbridge-id / slot / port | port-channel number } [ vrf vrf-name ] [ force-delete | no-refresh ]
clear ipv6 neighbor ipv6-address interface ve vlan_id [ vrf vrf-name ] [ force-delete | no-refresh ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### force-delete

Forcibly clears the cache.

### no-refresh

Prevents the cache from being refreshed.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

### vrf vrf-name

Specifies a VRF instance.

### slot linecard-number

Specifies a linecard.

### ipv6-address

Specifies the IPv6 address of a neighbor in *A::B::C:D* format.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter **?** to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**ve**

Specifies a virtual Ethernet (VE) interface.

*vlan\_id*

VLAN ID of the VE interface. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

## Modes

Privileged EXEC mode

## Examples

The following example clears the entire IPv6 neighbor discovery cache.

```
device# clear ipv6 neighbor
```

# clear ipv6 ospf

Clears OSPFv3 data processes, counts, force-spf, neighbors, redistribution, routes, and traffic.

## Syntax

```
clear ipv6 ospf all [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
clear ipv6 ospf counts [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
clear ipv6 ospf counts neighbor A.B.C.D [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
clear ipv6 ospf [ counts ] neighbor interface <N>gigabitethernet [ rbridge-id / ] slot / port [ A.B.C.D ]
clear ipv6 ospf [ counts ] neighbor interface { loopback number | port-channel number | ve vlan_id } [ A.B.C.D ] [ rbridge-id { rbridge-id | all } ]
clear ipv6 ospf { force-spf | redistribution | traffic } [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
clear ipv6 ospf neighbor all [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
clear ipv6 ospf routes { IPv6addr | all } [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
```

## Command Default

Disabled.

## Parameters

### all

Clears all OSPFv3 data.

### vrf vrf-name

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

### rbridge-id

(Optional) Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### counts

Clears OSPFv3 counters.

### neighbor

Clears all OSPF counters for neighbors.

### A.B.C.D

Specifies an IPv6 address.

### interface

Specifies an interface.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

**loopback** *number*

Specifies a loopback port number. The range is from 1 through 255.

**port-channel** *number*

Specifies a port-channel. The range is from 1 through 6144.

**ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface. Refer to the Usage Guidelines.

**force-spf**

Performs the shortest path first (SPF) calculation without clearing the OSPFv3 database.

**redistribution**

Clears OSPFv3 redistributed routes.

**traffic**

Clears OSPFv3 traffic statistics.

**neighbor**

Clears OSPFv3 neighbors.

**routes**

Clears OSPFv3 routes.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **force-spf** keyword to perform the shortest path first (SPF) calculation without clearing the OSPFv3 database.

If the physical interface type and name are specified, the **rbridge-id***bridge-id* option is not available.

On the Extreme VDX family of switches, VLANs are treated as interfaces from a configuration point of view. By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). Valid VLAN IDs are as follows:

- On Extreme VDX 8770 switches: 1 through 4086 for 802.1Q VLANs (VLAN IDs 4087 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.
- On all other Extreme VDX switches: 1 through 3962 for 802.1Q VLANs (VLAN IDs 3963 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.

## Examples

The following example restarts the OSPFv3 processes.

```
device# clear ipv6 ospf all
```

clear ipv6 ospf

The following example clears all OSPFv3 counters for a specified neighbor.

```
device# clear ipv6 ospf counts neighbor 10.10.10.1
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.0.0	This command was modified to support port-channels.



# clear ipv6 route

Clears IPv6 routing tables on an interface or line card and reloads the current information.

## Syntax

```
clear ipv6 route [ ipv6address/prefix ] [ rbridge-id { all | rbridge-id } | vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
clear ipv6 route all [ import [ src-vrf-name ] [ rbridge-id { rbridge-id | all } | vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
clear ipv6 route slot line_card_number [ ipv6address/ipv6prefix ] [ rbridge-id { rbridge-id | all } | vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*ipv6address/prefix*

IPv6 address and prefix in A:B::C:D/length format.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges in the cluster.

**vrf** *vrf-name*

Specifies a VRF instance from which the user is currently retrieving routes.

**import**

Specifies imported IPv6 routes.

*src-vrf-name*

Specifies a VRF instance from which routes are leaked.

**slot** *line\_card\_number*

Specifies a line card.

## Modes

Privileged EXEC mode

## Usage Guidelines

A standard prefix is required (for example, 2001:db8::/32, 2001:db8::/28, 2001:db8:1:1::/64).

## Examples

To clear the IPv6 route specified by the prefix 2001:db8::/32:

```
device# clear ipv6 route 2001:db8::/32
```

clear ipv6 route

## History

Release version	Command history
6.0.0	This command was modified to support the <b>vrf</b> keyword.
6.0.1a	This command was modified to support the <b>import</b> keyword.

# clear ipv6 vrrp statistics

Clears IPv6 VRRPv3 session statistics for all virtual groups, for a specified interface or RBridge ID, or for a specified virtual group.

## Syntax

```
clear ipv6 vrrp statistics [ all ]
```

```
clear ipv6 vrrp statistics interface { <N>gigabitethernet [ rbridge-id / ] slot / port | port-channel number | ve vlan_id }
```

```
clear ipv6 vrrp statistics [ session VRID | all ] [ rbridge { rbridge-id | all } ]
```

## Parameters

### all

Clears all IPv6 VRRP statistics.

### session *VRID*

Specifies the virtual group ID on which to clear statistics. The range is from 1 through 128.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

### interface

Specifies an interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

(Optional) Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel. The range is from 1 through 6144.

### ve *vlan\_id*

Specifies the VE VLAN number.

clear ipv6 vrrp statistics

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported in IPv6 VRRPv3 and VRRP-E-v3.

## Examples

The following example clears all IPv6 VRRPv3 statistics for all virtual groups.

```
device# clear ipv6 vrrp statistics all
```

The following example clears statistics for an IPv6 VRRPv3 session of virtual group 25.

```
device# clear ipv6 vrrp statistics session 25
```

The following example clears IPv6 VRRPv3 statistics on a specified port-channel.

```
device# clear ipv6 vrrp statistics interface port-channel 10
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

## Related Commands

[show ipv6 vrrp](#)

# clear lacp

Clears the Link Aggregation Group Control Protocol (LACP) counters on a specific port-channel.

## Syntax

```
clear lacp number counters
```

## Parameters

*number*

Specifies the port channel-group number. Valid values range from 1 through 6144.

**counters**

Clears traffic counters.

## Modes

Privileged EXEC mode

## Examples

To clear the LACP counters for a specific port-channel:

```
device# clear lacp 10 counters
```

## Related Commands

[show lacp](#)

# clear lacp counters

Clears the Link Aggregation Group Control Protocol (LACP) counters on all port-channels.

## Syntax

```
clear lacp counters
```

## Modes

Privileged EXEC mode

## Examples

To clear the counters for all port-channels:

```
device# clear lacp counters
```

## Related Commands

[show lacp](#)

# clear lldp neighbors

Clears the Link Layer Discovery Protocol (LLDP) neighbor information on all or specified interfaces.

## Syntax

```
clear lldp neighbors interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### interface

Use this parameter followed by the slot or port number to identify the interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *N*gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the **interface** parameter is not specified, this command clears the LLDP neighbor information received on all the interfaces.

## Examples

To clear the LLDP neighbor information for all interfaces:

```
device# clear lldp neighbors
```

## Related Commands

[show lldp neighbors](#)

# clear lldp statistics

Clears LLDP statistics for all interfaces or a specified interface.

## Syntax

```
clear lldp statistics interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### interface

Use this parameter followed by the slot or port number to identify the interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *N*gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the **interface** parameter is not specified, this command clears all the LLDP statistics on all interfaces.

## Examples

To clear all the LLDP statistics for all interfaces:

```
device# clear lldp statistics
```

## Related Commands

[show lldp statistics](#)



# clear logging auditlog

Clears the audit log system messages.

## Syntax

```
clear logging auditlog [ rbridge-id { rbridge-id | all } ]
```

## Command Default

This command is executed on the local switch.

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

## Examples

To clear the audit log messages on the local switch:

```
device# clear logging auditlog
```

## Related Commands

[clear logging raslog](#), [log-dampening-debug](#), [show logging auditlog](#)

# clear logging raslog

Clears RASLog messages from the switch.

## Syntax

```
clear logging raslog [ message-type { DCE | SYSTEM } ][ rbridge-id { rbridge-id | all } ]
```

## Command Default

Clear all RASLog messages on the local switch.

## Parameters

### message-type

Clears RASLog messages of the specified type.

#### SYSTEM

Clears system messages.

#### DCE

Clears DCE application messages.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

#### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

The **rbridge-id** operand is supported in VCS mode only.

This command is not supported on the standby management module.

## Examples

To clear all RASLog messages on the local switch:

```
switch# clear logging raslog
```

```
DCE Raslogs are cleared
SYSTEM Raslogs are cleared
```

To clear all DCE messages on the local switch:

```
switch# clear logging raslog message-type DCE
```

```
DCE Raslogs are cleared
```

To clear all SYSTEM messages on the local switch:

```
switch# clear logging raslog message-type SYSTEM
```

```
SYSTEM Raslogs are cleared
```

## Related Commands

[logging raslog console](#), [show logging raslog](#), [show running-config logging](#)

# clear mac-address-table conversational

Clears the conversational MAC interface status and configuration information.

## Syntax

```
clear mac-address-table conversational [ address mac_address | interface <N>gigabitethernet rbridge-id/slot/port | linecard
linecard_number [rbridge-id rbridge-id ] | vlan vlan_id ]
```

## Parameters

**address** *mac\_address*

Specifies a MAC address in HHHH.HHHH.HHHH format.

**interface**

Specifies an interface.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**linecard** *linecard\_number*

Specifies a line card on the local RBridge.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**vlan** *vlan\_id*

Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## History

Release version	Command history
5.0.0	This command was introduced.

# clear mac-address-table dynamic

Clears the dynamic MAC interface status and configuration information.

## Syntax

```
clear mac-address-table dynamic [ address mac_address | interface <N>gigabitethernet rbridge-id/slot/port | non-
authenticated | vlan vlan_id ]
```

## Parameters

**address** *mac\_address*

Specifies a MAC address in HHHH.HHHH.HHHH format.

**interface**

Specifies an interface.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**non-authenticated**

Specifies to clear all MAC addresses whose authentication has been denied by the authentication server (RADIUS server). This option is used to give the denied MAC addresses another chance to reauthenticate.

**vlan** *vlan\_id*

Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## History

Release version	Command history
5.0.0	This command description was modified to distinguish it from the form with the <b>conversational</b> keyword, and new keywords were added.
7.1.0	This command was modified to add <b>non-authenticated</b> option.

# clear maps dashboard

Clears the information from the Monitoring and Alerting Policy Suite (MAPS) dashboard log.

## Syntax

```
clear maps dashboard [ rbridge { rbridge-id | all }
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported in VCS fabric mode..

## Examples

Typical command example:

```
device# clear maps dashboard
```

## History

Release version	Command history
7.0.0	This command was introduced.

# clear nas statistics

Clears automatic network attached storage (NAS) statistics.

## Syntax

```
clear nas statistics all | server-ip ip_addr/prefix [ vlan vlan_id | vrf VRF_name ] [ rbridge-id rbridge-id ]
```

## Parameters

### all

Shows all gathered statistics.

### server-ip

IP address for which to clear Auto-NAS statistics.

*ip\_addr/prefix*

IPv4 address/prefix of a specified **Auto-** NAS port.

**vlan** *vlan\_id*

Specifies a VLAN interface for which to clear the statistics.

**vrf** *VRF\_name*

Specifies an OSPF VRF interface for which to clear the statistics.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID for which to clear the statistics.

## Modes

Privileged EXEC mode

## Examples

```
device# clear nas statistics all server-ip 1.1.1.0/24
```

# clear openflow

Clears a single OpenFlow rule based on a Flow ID or deletes all flows/groups/meters configured in the system.

## Syntax

```
clear openflow all | flowid flowid
```

## Parameters

**all**

Deletes all flows in the flow table, including group and meter-related configurations.

**flowid** *flowid*

Deletes a single OpenFlow rule with the specified Flow ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Examples

To delete a single OpenFlow rule based on a Flow ID:

```
device# clear openflow flowid 255
```

To delete all flows/groups/meters configured in the system:

```
device# clear openflow all
```

## History

Release version	Command history
6.0.1	This command was introduced.
6.0.1a	This command was modified to enhance the description of the <b>all</b> keyword.



# clear overlay-gateway

Clear counters for the specified gateway.

## Syntax

```
clear overlay-gateway name { statistics | vlan statistics }
```

## Parameters

*name*

Specifies the name of the VXLAN gateway profile.

**statistics**

Clears all statistics for the VXLAN gateway.

**vlan statistics**

Clears per-VLAN statistics for the VXLAN gateway.

## Modes

Privileged EXEC mode

## Usage Guidelines

If you specify the VXLAN gateway name, the gateway must already be configured.

If you specify VLAN IDs, these VLANs must already be configured as exported VLANs for the gateway.

## Examples

The following example clears all counters for the already configured VXLAN gateway named gateway1.

```
device# clear overlay-gateway gateway1 statistics
```

# clear policy-map-counters

Provides a mechanism for clearing the policy map counters.

## Syntax

```
clear policy-map-counters [ interface <N>gigabitethernet rbridge-id/slot/port | port-channel number ]
```

## Parameters

### interface

Specifies an interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

### port-channel *number*

Specifies the interface is a port-channel. The number of available channels range from 1 through 6144.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command with a specific interface and direction to clear the policy map counters for that interface.

Use this command without identifying an interface and direction of traffic to clear all of the policy map counters..

## Examples

To clear the policy map counters for a specific interface use the following command:

```
device# clear policy-map-counters interface tengigabitethernet 2/0/2
```

# clear sessions

Logs out the user sessions connected to the switch.

## Syntax

```
clear sessions [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is not distributed across the cluster. The RBridge ID of the node should be used to log out users connected to the individual nodes.

The **rbridge-id** operand is supported in VCS mode only.

## Examples

```
device# clear sessions rbridge-id 3
This operation will logout all the user sessions. Do you want to continue (yes/no)? : y
```

# clear sflow statistics

Clears sFlow statistics from all ports or from a specified port..

## Syntax

```
clear sflow statistics interface [ <N>gigabitethernet rbridge-id/slot/port | tunnel ]
```

## Parameters

### interface

Specifies an interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### tunnel

Specifies the tunnel interface.

## Modes

Privileged EXEC mode

## Examples

To clear sFlow statistics:

```
device# clear sflow statistics
```

# clear spanning-tree counter

Clears all spanning-tree counters on the interface.

## Syntax

```
clear spanning-tree counter [ interface | port-channel number | <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### interface

Specifies an interface.

### port-channel *number*

Specifies the interface is a port-channel. The number of available channels ranges from 1 through 6144.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace \ **N**gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the **interface** parameter is not specified, spanning-tree counters are cleared for all interfaces.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

## Examples

To clear spanning-tree counters for all interfaces:

```
device# clear spanning-tree counter
```

To clear spanning-tree counters for a 10-gigabit Ethernet interface:

```
device# clear spanning-tree counter interface tengigabitethernet 0/1
```

clear spanning-tree counter

To clear spanning-tree counters for port-channel 23:

```
device# clear spanning-tree counter interface port-channel 23
```

## Related Commands

[show spanning-tree](#)

# clear spanning-tree detected-protocols

Clears all spanning-tree detected protocols on the interface.

## Syntax

```
clear spanning-tree detected-protocols [ interface | port-channel number | <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### interface

Specifies an interface.

### port-channel *number*

Specifies the interface is a port-channel. The number of available channels ranges from 1 through 6144.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *N*gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the **interface** parameter is not specified, spanning-tree detected protocols are cleared for all interfaces.

## Examples

To clear detected protocols for all interfaces:

```
device# clear spanning-tree detected-protocols
```

To clear detected protocols for a 10-gigabit Ethernet interface:

```
device# clear spanning-tree detected-protocols interface tengigabitethernet 0/1
```

To clear detected protocols for port-channel 23:

```
device# clear spanning-tree detected-protocols interface port-channel 23
```

clear spanning-tree detected-protocols

## Related Commands

[show spanning-tree](#)



# clear statistics openflow

Clears the flow statistics for all flows or for a specified flow.

## Syntax

```
clear statistics openflow [controller | flow | group | meter]
```

## Parameters

*controller*

Sends statistics for the controller in a flow.

*flow*

Deletes the flow statistics for a specified flow on the OpenFlow controller.

*group*

Clears statistics for all groups.

*meter*

Clears statistics for all meters.

## Modes

Privileged EXEC mode

## Examples

```
device# clear statistics openflow
Possible completions:
 controller  send to controller statistics
 flow        Flow
 group       Clear statistics for all groups
 meter       Clear statistics for all meters
 |           Output modifiers
device#
```

## History

Release version	Command history
6.0.1	This command was introduced.

# clear support

Removes support data such as core files and RAS FFDC files from the switch.

## Syntax

```
clear support [ rbridge-id { rbridge-id | all } ]
```

## Command Default

This command is executed on the local switch.

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

The **rbridge-id** operand is supported in VCS mode only.

## Examples

To remove core files from the local switch:

```
switch# clear support
```

## Related Commands

[copy support](#), [show support](#)

# clear udd statistics

Clears UDLD statistics.

## Syntax

```
clear udd statistics [ interface { <N>gigabitethernet rbridge-id/slot/port } ]
```

## Parameters

### interface

Specifies an interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Clears either all unidirectional link detection (UDLD) protocol statistics or clears the statistics on a specified port.

## Examples

To clear UDLD statistics on a specific tengigabitethernet interface:

```
device# clear udd statistics interface te 5/0/1
```

## Related Commands

[protocol udd](#), [show udd](#), [udd enable](#)

# clear vrrp statistics

Clears VRRP statistics.

## Syntax

**clear vrrp statistics**

**clear vrrp statistics interface** { <N>**gigabitethernet** [ *rbridge-id* / ] *slot* / *port* | **port-channel** *number* | **ve** *vlan\_id* }

**clear vrrp statistics session** *VRID*

## Parameters

### interface

Specifies an interface.

### <N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

(Optional) Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

### **ve** *vlan\_id*

Specifies the VE VLAN number.

### **port-channel** *number*

Specifies a port-channel. The range is from 1 through 6144.

### **session** *VRID*

Specifies the virtual group ID on which to clear statistics. The range is from 1 through 255.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command clears VRRP session statistics for all virtual groups, for a specified interface or for a specified virtual group.

This command is for VRRP and VRRP-E. VRRP-E supports only the **ve** *vlan\_id* type.

To clear all vrrp statistics, use the **clear vrrp statistics** command with no operands.

## Examples

The following example clears all VRRP statistics for all virtual groups.

```
device# clear vrrp statistics
```

The following example clears statistics for a 10-gigabit Ethernet interface that has an rbridge-id/slot/port of 121/0/50.

```
device# clear vrrp statistics interface tengigabitethernet 121/0/50
```

The following example clears statistics for a session for a VRRP virtual group called "vrrp-group-25".

```
device# clear vrrp statistics session 25
```

The following example clears VRRP statistics on a specified port-channel.

```
device# clear vrrp statistics interface port-channel 10
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

## Related Commands

[show vrrp](#)

# CLI

In a Python shell, runs a device CLI command or series of commands. You can also assign the output of such commands to a Python object.

## Syntax

```
CLI ( ' device-CLI-command ' [ \n ' device-CLI-command ' ] [ do_print = ] { True | False } )
```

## Parameters

*device-CLI-command*

A Network OS CLI command. You separate additional commands with `\n`.

**do\_print =**

Specify whether or not to print the output of *device-CLI-command* to the default device. The default is to print the output.

**True**

Print the output.

**False**

Do not print the output.

## Modes

Python command shell

## Usage Guidelines

Divergences between the CLI syntax and Python syntax include the following differences:

- Although in general, the CLI syntax is not case-sensitive, our convention is to use lower-case.
- Python syntax is case sensitive. Regarding the syntax documented in the current topic, note the following:
  - The syntax of the command is upper case (CLI) and not lower case (cli).
  - The syntax of the **do\_print =** options is to capitalize the first letter: { **True** | **False** }

In Python, double quotes (") and single quotes (') are equivalent.

As delimiter between multiple CLI commands, use `\n`.

There is a difference between running a sequence of Network OS CLI commands in the Python shell rather than in the standard Network OS interface. Whereas in the standard interface the result of a command is persistent, in the Python shell each `CLI ( )` statement is independent of any preceding ones.

For support of the `CLI ( )` command, although a Python script must include a `from CLI import CLI` statement, this statement is automatically implemented when launching the Python interpreter interactively.

Within a script or interactive session, if you assign a CLI command or series of commands to a Python variable, you can then append the following functions to the variable:

- **.rerun()**—updates the variable from a new run of the CLI command or series of commands.

```
device# python
Python 3.3.2 (default, Apr 11 2014, 13:05:18)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> cmd_show_running_vlan = CLI('show running-config interface vlan')
# The Network OS show running-config int vlan command is run,
# and that command is assigned to the Python variable cmd_show_running_vlan.
!Command: show running-config interface vlan
!Time: Tue Jan 6 00:12:37 2015
interface Vlan 1
!
>>> cmd_config_vlans = CLI('configure \n interface vlan 101-103 \n exit')
# A series of three commands are run and assigned to the Python variable
cmd_config_vlans.
!Command: configure \n interface vlan 101-103 \n exit
!Time: Tue Jan 6 00:24:17 2015
>>> cmd_show_running_vlan.rerun()
# The rerun() function appended to cmd_show_running_vlan gives the following output:
!Command: show running-config interface vlan
!Time: Tue Jan 6 00:24:34 2015
interface Vlan 1
!
interface Vlan 101
!
interface Vlan 102
!
interface Vlan 103
!
```

- **.get\_output()**—returns the value of a new run of the CLI command or series of commands, as a list.

```
#Required in all scripts for NOS:
from CLI import CLI

# Import the Python Regular Expressions (re) module:
import re

# Create Python objects:
rbridges = []
wwns = []
mgmt_ip_addresses = []
vcs_status = []
fabric_status = []
hostnames = []

cmd_show_vcs = CLI("show vcs", False)

# Using .get_output(), assign the result of show vcs to a Python object named output:
output = cmd_show_vcs.get_output()

for line in output:
    found = re.search(r'^(\d+)\s+(\S+)\s+(\S+)\s+(\S+)\s+(\S+)\s+(\S+)\s+(\S+)\s+(\S+)\s+(\S+)\$', line, re.M)

    if found:
        rbridges.append(found.group(1))
        wwns.append(found.group(2).strip('>'))
        mgmt_ip_addresses.append(found.group(3))
        vcs_status.append(found.group(4))
        fabric_status.append(found.group(5))
        hostnames.append(found.group(6))

print('rbridges: ', rbridges)
print('wwns: ', wwns)
print('mgmt_ip_addresses: ', mgmt_ip_addresses)
print('vcs_status: ', vcs_status)
```

```
print('fabric_status: ', fabric_status)
print('hostnames: ', hostnames)
```

## Examples

The following example launches the Python shell and then both assigns a series of CLI configuration commands to a Python variable and runs those commands.

```
device# python
Python 3.3.2 (default, Apr 11 2014, 13:05:18)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> cmd_config_vlans = CLI('config \n int vlan 101-103 \n exit')
!Command: config
  int vlan 101-103
  exit
!Time: Tue Jan  6 00:24:17 2015
>>>
```

The following example launches the Python shell and then both assigns a CLI operational command to a Python variable and runs that command.

```
device# python
Python 3.3.2 (default, Apr 11 2014, 13:05:18)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> cmd_reload_system = CLI('reload system \n y')
```

## History

Release version	Command history
6.0.1	This command was introduced.



# client-to-client-reflection

Enables routes from one Route Reflector (RR) client to be reflected to other clients by the host device on which it is configured.

## Syntax

`client-to-client-reflection`

`no client-to-client-reflection`

## Command Default

Enabled.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

When this command is used, the host device on which it is configured becomes the route-reflector server.

The **no** form of the command disables route reflection between clients.

## Examples

The following example configures client-to-client reflection on the BGP host device for the IPv4 unicast address-family.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# client-to-client-reflection
```

The following example disables client-to-client reflection on the BGP host device for the IPv6 unicast address-family.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# no client-to-client-reflection
```

The following example configures client-to-client reflection in L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# client-to-client-reflection
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Usage guidelines were modified to indicate inheritance from the default VRF in support of Multi-VRF.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.

## Related Commands

[address-family l2vpn evpn](#), [address-family unicast \(BGP\)](#)

# clock set

Sets the local clock date and time.

## Syntax

```
clock set CCYY-MM-DDTHH:MM:SS [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*CCYY-MM-DDTHH:MM:SS*

Specifies the local clock date and time in year, month, day, hours, minutes, and seconds. Valid date and time settings range from January 1, 1970 to January 19, 2038.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

An active NTP server, if configured, automatically updates and overrides the local clock time. The RBridge ID of the node should be used to set the clock.

## Examples

The following example sets the date and time to March 17, 2010, 15 minutes past noon for all switches in the cluster.

```
device# clock set rbridge-id all 2010-03-17T12:15:00
```

## Related Commands

[clock timezone](#) (Privileged EXEC mode), [ntp server](#), [show clock](#)

# clock timezone (Privileged EXEC mode)

Sets the time zone based on region and longitudinal city.

## Syntax

```
clock timezone region/city [ rbridge-id { rbridge-id | all } ]
```

```
no clock timezone [ rbridge-id rbridge-id ]
```

## Parameters

*region*

Specifies the region's time zone.

*city*

Specifies the city's time zone.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Sets the local clock time zone.

Regions include the following countries: Africa, America, Pacific, Europe, Antarctica, Asia, Australia, Atlantic, Indian, and longitudinal city. For a complete listing of supported regions and cities, refer to time zone appendix in the *Network OS Administrator's Guide* .

By default, all switches are in the Greenwich Mean Time (GMT) time zone. The **no** operand removes the time zone setting for the local clock. When using the **no** operand, you do not need to reference a time zone setting.

The **no** operand is not distributed across the cluster. The RBridge ID of the node should be used.

Network Time Protocol (NTP) commands must be configured on each individual switch.

The region name and city name must be separated by a slash (/).

After upgrading your switch to a new Network OS version, you might need to reset the time zone information.

This command can also be run in RBridge ID configuration mode.

## Examples

To set the time zone to Pacific Standard Time in North America on all nodes in the cluster:

```
device# clock timezone America/Los_Angeles rbridge-id all
```

To remove the time zone setting:

```
device# no clock timezone rbridge-id 5
```

## Related Commands

[clock set](#), [ntp server](#), [show clock](#)

# clock timezone (RBridge ID configuration mode)

Sets the time zone based on region and longitudinal city.

## Syntax

**clock timezone** *region/city*

**no clock timezone**

## Parameters

*region*

Specifies the region's time zone.

*city*

Specifies the city's time zone.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Sets the local clock time zone.

Regions include the following countries: Africa, America, Pacific, Europe, Antarctica, Asia, Australia, Atlantic, Indian, and longitudinal city. For a complete listing of supported regions and cities, refer to Appendix C in the *Network OS Administrator's Guide*.

By default, all switches are in the Greenwich Mean Time (GMT) time zone. The **no** operand removes the timezone setting for the local clock. When using the **no** operand, you do not need to reference a timezone setting.

The **no** operand is not distributed across the cluster.

Network Time Protocol (NTP) commands must be configured on each individual switch.

The region name and city name must be separated by a slash (/).

Upgrade considerations: Existing timezone of system is retained after firmware upgrade, and it will be updated in configuration settings.

Downgrade considerations: Existing timezone of system will be retained after firmware downgrade and the respective entry will be removed from configuration settings.

This command can also be run in Privileged EXEC configuration mode.

## Examples

To set the time zone to Pacific Standard Time in North America on all nodes in the cluster:

```
device# configure terminal
device(config)# rbridge-10
device(config-rbridge-id-10# clock timezone America/Los_Angeles
```

To remove the timezone setting:

```
device# configure terminal
device(config)# rbridge-10
device(config-rbridge-id-10# no clock timezone
```

# cluster-id

Configures a cluster ID for the route reflector.

## Syntax

```
cluster-id { num | ip-addr }
```

```
no cluster-id { num | ip-addr }
```

## Command Default

The default cluster ID is the device ID.

## Parameters

*num*

Integer value for cluster ID. Range is from 1 through 65535.

*ip-addr*

IPv4 address in dotted-decimal notation.

## Modes

BGP configuration mode

## Usage Guidelines

When configuring multiple route reflectors in a cluster, use the same cluster ID to avoid loops within the cluster.

The **no** form of the command restores the default.

## Examples

The following example configures a cluster ID for the route reflector.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# cluster-id 1234
```



# compare-med-empty-**aspath**

Enables comparison of Multi-Exit Discriminators (MEDs) for internal routes that originate within the local autonomous system (AS) or confederation

## Syntax

```
compare-med-empty-aspath  
no compare-med-empty-aspath
```

## Command Default

Disabled.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command restores the default.

## Examples

The following example configures the device to compare MEDs.

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# router bgp  
device(config-bgp-router)# compare-med-empty-aspath
```

# compare-routerid

Enables comparison of device IDs, so that the path-comparison algorithm compares the device IDs of neighbors that sent otherwise equal-length paths.

## Syntax

```
compare-routerid  
no compare-routerid
```

## Modes

BGP configuration mode

## Examples

The following example configures the device always to compare device IDs.

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# router bgp  
device(config-bgp-router)# compare-routerid
```

# confederation identifier

Configures a BGP confederation identifier.

## Syntax

**confederation identifier** *autonomous-system number*

**no confederation identifier**

## Command Default

No BGP confederation identifier is identified.

## Parameters

*autonomous-system number*

Specifies an autonomous system number (ASN). The configurable range of values is from 1 through 4294967295.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command removes a BGP confederation identifier.

## Examples

The following example specifies that confederation 65220 belongs to autonomous system 100.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# local-as 65220
device(config-bgp-router)# confederation identifier 100
```

## History

Release version	Command history
5.0.0	This command was introduced.

# confederation peers

Configures subautonomous systems to belong to a single confederation.

## Syntax

**confederation peers** *autonomous-system number* [ ...*autonomous-system number* ]  
**no confederation peers**

## Command Default

No BGP peers are configured to be members of a BGP confederation.

## Parameters

*autonomous-system number*

Autonomous system (AS) numbers for BGP peers that will belong to the confederation. The configurable range of values is from 1 through 4294967295.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command removes an autonomous system from the confederation.

## Examples

The following example configures autonomous systems 65520, 65521, and 65522 to belong to a single confederation under the identifier 100.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# local-as 65020
device(config-bgp-router)# confederation identifier 100
device(config-bgp-router)# confederation peers 65520 65521 65522
```

## History

Release version	Command history
5.0.0	This command was introduced.

# configure terminal

Enters global configuration mode.

## Syntax

`configure terminal`

## Modes

Privileged EXEC mode

## Related Commands

`exit`

# conform-set-dscp

Configures the packet DSCP priority of a class map.

## Syntax

```
conform-set-dscp dscp-num
```

```
no conform-set-dscp dscp-num
```

## Parameters

*dscp-num*

Specifies that traffic with bandwidth requirements within the rate configured for CIR that has the packet DSCP priority set to the value specified by the *dscp-num* variable. Valid values are 0 through 63.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class map.

If the optional parameters for a class-map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

Use the **no** version of this command to remove the parameter from the class map.

## Examples

Example of setting this parameter.

```
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# conform-set-dscp 3
```

# conform-set-prec

Configures the packet IP precedence value of a class map.

## Syntax

```
conform-set-prec prec-num
```

## Parameters

*prec-num*

Specifies that traffic with bandwidth requirements within the rate configured for CIR will have packet IP precedence value (first 3 bits of DSCP) set to the value in the *prec-num* variable. Valid values are 0 through 7.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class map.

If the optional parameters for a class-map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

Use the **no** version of this command to remove the parameter from the class map.

## Examples

Example of setting this parameter.

```
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# conform-set-prec 3
```

# conform-set-tc

Configures the CIR internal queue assignment of a class map.

## Syntax

```
conform-set-tc trafficclass
```

```
no conform-set-tc trafficclass
```

## Parameters

*trafficclass*

Specifies that traffic with bandwidth requirements within the rate configured for CIR has the traffic class (internal queue assignment) set to the configured value. Valid values are 0 through 7.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class-map.

If the optional parameters for a class map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

Use the **no** version of this command to remove the parameter from the class map.

## Examples

Example of setting this parameter.

```
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# conform-set-tc 3
```



# connector-group

Designates which connector group that FlexPort is allowed to access on the switch.

## Syntax

```
connector-group rbridge-id/slot/group
```

## Command Default

The default connector group is undefined.

## Parameters

*rbridge-id/slot/port*

Specifies a valid Fibre Channel port interface

*rbridge-id*

Specifies the RBridge ID.

*slot*

Specifies a valid slot number.

*group*

Specifies a connector group on the switch. The connector-group numbers range from 1 through 8.

## Modes

Hardware configuration mode

## Usage Guidelines

This command is supported only on the Extreme VDX 2741, VDX 2746, VDX 6740, VDX 6740T, and VDX 6740T-1G .

The connector-group numbers are related directly to the ports as numbered on each platform. The connector-group numbers that are allowed to be changed and their associated port numbers are shown in the table below. For example, on an Extreme VDX 6740, ports 1 through 8 belong to connector group 1. Not every connector group is supported on a switch.

Changing the connector-group speed is disruptive to all ports within the group, depending on their configuration and the new connector-group to be used. A speed change is disallowed if any port in the connector group is already running at a speed that cannot be supported with the new connector group speed.

The Extreme VDX 2741 has eight ports on the second connector group. The ports from the second connector group does not support Fibre Channel.

**NOTE**

Issuing the **copy default-config startup-config** command does not erase the hardware connector configuration after a reboot.

**TABLE 6** Flexport supported hardware

Platform	Port number range	Connector group
Extreme VDX 6740V, 6740T, and VDX 6740T-1G	1-8	1
	17-24	3
	33-40	5
	41-48	6
Extreme VDX 2741	29-36	1
	37-44	2
Extreme VDX 2746	43-50	1
	51-56	2
Extreme VDX 6740T	49-50	7
Extreme VDX 6740T-1G	51-52	8

## Examples

This example sets the connector group for Rbridge-ID 1 to group 1.

```
device# configure terminal
device(config)# hardware
device(config-hw)# connector-group 1/0/1
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[flexport](#), [hardware](#), [speed \(FlexPort\)](#)

# connector

Executes connector mode for the purpose of configuring breakout mode on Quad SFPs (QSFPs).

## Syntax

```
connector rbridge-id/slot/port
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Hardware configuration mode

## Examples

```
device# configure terminal
device(config)# hardware
device(config-hardware)# connector 2/0/1
device(config-connector-2/0/1)#
```

## Related Commands

[sfp breakout](#), [power-off linecard](#), [power-on linecard](#)

# continue

Configures a route-map instance number that goes in a continue statement in a route-map instance.

## Syntax

**continue** *number*

**no continue** *number*

## Parameters

*number*

Route-map instance number. Range is from 1 through 4294967295.

## Modes

Route map configuration mode

## Related Commands

[route-map](#)

# controller

Specifies the global name of an OpenFlow controller in OpenFlow logical-instance configuration mode.

## Syntax

**controller** *name*

**no controller** *name*

## Parameters

**controller** *name*

Specifies the already-created name of an OpenFlow controller.

## Modes

OpenFlow logical-instance configuration mode

## Usage Guidelines

This command specifies the controller to be used as the active-mode controller for the specified logical instance. The logical instance must have been created by means of the global **openflow-controller** command. To remove an active-mode controller from the logical instance, use the **no** form of this command.

## Examples

The following example enters OpenFlow logical-instance configuration mode and specifies the name of an OpenFlow controller.

```
device(config)# rbridge-id 12
device(config-rbridge-id-12)# openflow logical-instance 1
device(config-logical-instance-1)# controller mycontroller
device(config-logical-instance-1)#
```

## History

Release version	Command history
6.0.1	This command was introduced.

# copy

Copies configuration data.

## Syntax

```
copy source_file destination_file
```

## Parameters

*source\_file*

The source file to be copied. Specify one of the following parameters:

**default-config**

The default configuration.

**global-running-config**

Global data of the running configuration.

**rbridge-running-configuration** *rbridge-id*

Running configuration of a specified RBridge.

**flash://filename**

A file in the local flash memory.

**NOTE**

This option is not supported on the Extreme VDX 2740 or Extreme VDX 2746.

**ftp://username:password@host\_ip\_address/path**

A file on a remote host. Transfer protocol is FTP.

**scp://username:password@host\_ip\_address/path**

A file on a remote host. Transfer protocol is SCP.

**sftp://username:password@host\_ip\_address/path**

A file on a remote host. Transfer protocol is SFTP.

**tftp://username:password@host\_ip\_address/path**

A file on a remote host. Transfer protocol is TFTP.

**usb://path**

A file on an attached USB device.

*destination\_file*

The destination file. Specify one of the following parameters:

**default-config**

The default configuration.

**global-running-config**

Global data of the running configuration.

**local-running-configuration**

Local data of the running configuration.

**rbridge-running-configuration** *rbridge-id*  
Running configuration of a specified RBridge.

**flash://filename**  
A file in the local flash memory.

**ftp://username:password@host\_ip\_address//path**  
A file on a remote host. Transfer protocol is FTP.

**scp://username:password@host\_ip\_address//path**  
A file on a remote host. Transfer protocol is SCP.

**sftp://username:password@host\_ip\_address/path**  
A file on a remote host. Transfer protocol is SFTP.

**tftp://username:password@host\_ip\_address/path**  
A file on a remote host. Transfer protocol is TFTP.

**usb://path**  
A file on an attached USB device.

**use-vrf** *vrf-id*  
Use this option to specify the name of the VRF where the host is located. If this option is not set, mgmt-vrf is used by default.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to back up and restore configuration files with various protocols.

This command is supported only on the local switch.

IPv4 and IPv6 addresses are supported.

The special characters of dollar sign "\$" and exclamation point "!" can be used as part of the password variable, provided they are paired with the correct escape characters. The "\$" must be paired with two backslashes "\". For example, if your password choice was "\$password" on a remote server, you must use "username:\\\$password@1.1.1.1" for the **copy** command. The exclamation point must be paired with a single backslash in the **copy** command, such as "username:\\!password@1.1.1.1".

## Examples

To save the running configuration to a file:

```
device# copy running-config flash://myconfig
```

To overwrite the startup configuration with a locally saved configuration file:

```
device# copy flash://myconfig running-config
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.



# copy snapshot

Uploads and downloads configuration snapshot files to and from an FTP or SCP server.

## Syntax

```
copy snapshot rbridge-id rbridge-id snapshot-id snapshot-id ftp://directory_path
copy snapshot rbridge-id rbridge-id snapshot-id snapshot-id scp://directory_path
copy snapshot ftp:// directory_path rbridge-id rbridge-id snapshot-id snapshot-id
copy snapshot scp:// directory_path rbridge-id rbridge-id snapshot-id snapshot-id
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies the RBridge ID whose configuration snapshot has been captured.

**snapshot-id** *snapshot-id*

Specifies the name of the snapshot that has been captured.

*directory\_path*

Specifies the FTP or SCP directory path to which you are uploading the snapshot or from which you are downloading the snapshot.

## Modes

Privileged EXEC mode

## Usage Guidelines

If a snapshot was taken on a node that had been disconnected from the cluster, the cluster will not have the snapshot. Therefore, you can use these commands to upload the snapshot from the disconnected RBridge ID to an FTP or SCP server, then download it to an RBridge ID on the cluster.

### NOTE

The uploaded snapshot configuration file is stored as a tar file (of the form *rbridgid-snapshotID*) on the FTP or SCP server.

## Examples

To upload a snapshot configuration file called node4configuration to an FTP server:

```
device# copy snapshot rbridge-id 11 snapshot-id node4configuration ftp://backupdir_path
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

copy snapshot

## Related Commands

[vcs config snapshot](#)

# copy support

Copies support data to a remote host or a USB device.

## Syntax

```
copy support { ftp | scp | support-param | usb } user user_name group group_name password password host
ip_address linecard linecard_string directory dir [ sub-directory dir ] [ timeout multiplier ] [ rbridge-id { rbridge-id | all }
[ use-vrf vrf-name ]
```

## Parameters

**ftp | scp | usb**

Specifies the File Transfer Protocol (ftp), the Secure Copy Protocol (scp), or the USB directory.

**support-param**

Enables specification of an optional subdirectory for uploading copy support files.

**user** *user\_name*

Specifies the user login name for the server.

**group** *group\_name*

Specifies the group login name for the server. As many as four group names, separated by commas, can be specified.

**password** *password*

Specifies the account password.

**host** *host\_ip*

Specifies the host IP address in IPv4 or IPv6 format.

**linecard** *linecard\_string*

Specifies the line card to upload support data. Lx <x=1-4 on M4 platforms, x=1-8 on M8 platforms>.

**directory** *dir*

*Specifies a fully qualified path* to the directory where the support data will be stored.

**subdirectory** *dir*

*Specifies a fully qualified path* to the subdirectory where the support data will be stored. (Refer to the Usage Guidelines.)

**timeout** *multiplier*

Specifies a timeout multiplier. Valid multipliers are 1 through 5. When a timeout multiplier is specified, the default timeout value for each module is multiplied by the specified timeout value.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**all**

Specifies all switches in the fabric.

**use-vrf** *vrf-name*

Specifies a VRF through which to communicate with the server. If this option is not set, mgmt-vrf is used by default.

## Modes

Privileged EXEC mode

## Usage Guidelines

The support data is saved in the following format:

```
switchname-IPaddress-slotnumber-cputype-timestamp.modulename.txt.ss.gz
```

Example: sw0-10.123.10.5-S5cp-201204081630.OS.txt.ss.gz

Pagination is not supported with this command. Use the "more" parameter to display the output one page at a time.

The subdirectory is appended to the copy support main directory, which is stored as a Distributed Configuration Manager (DCM) configuration item. DCM supports the configuration management of multinode cluster applications and clustering for VCS.

## Examples

To save support data on an attached USB device:

```
device# usb on
USB storage enabled
switch# copy support usb directory support
```

To copy support data to a subdirectory:

```
device# copy support support-param sub-directory M8 timeout 3
```

## Related Commands

[clear logging raslog](#), [copy support-interactive](#), [show support](#)

# copy support-interactive

Copies support data interactively.

## Syntax

**copy support-interactive**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is functionally equivalent to the **copy support** command.

Answering **Y** to the Extreme VCS Fabric support prompt indicates that your switch is in Extreme VCS Fabric mode. Support data will be copied from all nodes in the fabric.

The interactive command interface prompts you for the following information:

- Server Name or IP Address (IPv4 only)
- Protocol (FTP or SCP)
- User login name
- Password
- Directory
- Rbridge ID
- Module timeout multiplier
- **use-vrf**

## Examples

To upload support data interactively:

```
device# copy support-interactive
Save to USB device [y/n]: n
Server Name or IP Address: 10.30.33.131
Protocol (ftp, scp): ftp
User: admin
Password: *****
Directory: /home/admin/support
Enter 'all' for all nodes or specify the rbridgeId(s) of the node(s) [Default: Local
Node]: all
Module timeout multiplier[Range:1 to 5.Default:1]: 2
Enter VRF name [mgmt-vrf]:vrf-name

Rbridge-id 195: Saving support information for chassis:sw0, module:RAS...
(output truncated)
```

## History

Release version	Command history
7.0.0	This command entry was enhanced.

## Related Commands

[clear support](#), [show copy-support status](#)

# cos-mutation

Specifies the mutation-map to be used on the port.

## Syntax

```
cos-mutation map_name
```

## Parameters

*map\_name*

The user-defined map-name.

## Modes

Policy-map configuration mode

## Usage Guidelines

This command is allowed only for the Ingress direction.

This command can only be configured in for the **class class-default** command.

This command can lead to a possible contradiction if there are other user-defined classes used in the same policy-map which has a set CoS action configured. In this case, the defined CoS takes priority over the mutation map.

## Examples

Typical command example:

```
device# configure terminal
device(config)# policy-map mutation
device(config-policymap)# class class-default
device(config-policyclass)# cos-mutation plsmap
```

## Related Commands

[class](#), [policy-map](#)

# counter reliability

Under Access Gateway, sets the reliability counter for the N\_Port Monitoring feature.

## Syntax

**counter reliability** *value*

## Parameters

*value*

Specifies from 10 through 100 static change notifications (SCNs) per 5-minute period. The default is 25 SCNs.

## Modes

Access Gateway (AG) configuration mode

## Usage Guidelines

The command sets and displays the reliability count of online and offline SCNs counted during a 5-minute period before the link between an N\_Port on a VDX switch in Access mode and an F\_Port on an FC fabric is considered unreliable.

## Examples

The following example sets the reliability counter value.

```
device# configure terminal
device(config)# rbridge 3
device(config-rbridge-id-3)# ag
device(config-rbridge-id-3-ag)# counter reliability 50
```

## History

Release version	Command history
6.0.1	This command is available only as an independent command, in AG configuration mode. Previously, in RBridge-ID configuration mode it could be executed with the <b>ag</b> prefix.

## Related Commands

[ag](#)



# crypto ca authenticate

Downloads the CA certificate from the remote certificate server for the trust point.

## Syntax

```
crypto ca authenticate { trustpointCA_name directory remote_dir_name file cert_file host host_address protocol {FTP | SCP}
  user host_login password host_user_password [rbridge-id {rbridge-id | all}]}
```

```
no crypto ca authenticate { trustpointCA_name [rbridge-id {rbridge-id | all}]}
```

## Parameters

*trustpointCA\_name trustpointCA\_name*

Defines the name of the trust point you are authenticating. This name needs to be the same as that of the trust point created by the **crypto ca trustpoint** command. The string for the name can not be left blank. The length of the string can range from 1 through 64 characters.

**directory** *remote\_dir\_name*

Defines the directory where the certification file resides.

**file** *cert\_file*

Defines the name of the certification file.

**host** *host\_address*

Defines the host name or IP address of the remote certificate server.

**protocol** {FTP | SCP}

Specifies the use of either FTP or SCP protocol for accessing the certification file.

**user** *host\_login*

Defines user name for the host server.

**password** *host\_user\_password*

Defines the password for the user name on the host server.

### NOTE

It is recommended to not list the password in command line for security purposes; the user will be prompted for the password.

**rbridge-id** {*rbridge-id* | all}

If unspecified, executes only for current node. If a particular rbridge-id is specified then the command is executed for that node. If **rbridge-id all** is specified, the command executes for all nodes in the cluster.

## Modes

Privileged EXEC mode

## Usage Guidelines

This is the CA certificate of the Trusted CA that you want to sign the CSR and generate the identity certificate.

The *trustpoint\_CAname* name needs to be the same as that of the trust point created by the **crypto ca trustpoint** command.

The **no** form of the command deletes the certificate.

## Examples

Typical command example.

```
device# crypto ca authenticate t1 protocol SCP host 10.70.12.102 user fvt directory /users/home/crypto
file cacert.pem
Password: *****
```

## History

Release version	Command history
6.0.0	This command was introduced.

# crypto ca enroll

Enrolls the trust point by generating the Certificate Signing Request (CSR) and exporting it to the remote certificate server.

## Syntax

```
crypto ca enroll { trustpointCA_name directory remote_dir_name host host_address protocol {FTP | SCP} user host_login
password host_user_password country country state state locality locality organization organization orgunit orgunit
common common_name[rbridge-id {rbridge-id | all}]}
```

## Parameters

*trustpointCA\_name*

Defines the name of the trust point you are enrolling. This name needs to be the same as that of the trust point created by the **crypto ca trustpoint** command. The string for the name can not be left blank. The length of the string can range from 1 through 64 characters.

**directory** *remote\_dir\_name*

Defines the path of the directory to export the Certificate Signing Request.

**host** *host\_address*

Defines the host name or IP address of the remote certificate server.

**protocol** {FTP | SCP}

Specifies the use of either FTP or SCP protocol for exporting the certification file.

**user** *host\_login*

Defines user name for the host server.

**password** *host\_user\_password*

Defines the password for the user name on the host server.

### NOTE

It is recommended to not list the password in command line for security purposes; the user will be prompted for the password.

**country** *country*

Defines the two-letter country code for generating the CSR.

**state** *state*

Defines the state name for generating the CSR.

**locality** *locality*

Defines the locality name for generating the CSR.

**organization** *organization*

Defines the organizational unit name for generating the CSR.

**orgunit** *orgunit*

Defines the name of the certification file.

**common** *common\_name*

This is the name used to connect to the device through HTTPS. Enter a Fully Qualified Domain Name (FQDN) or IP address. If a FQDN is used, you need to configure a domain name and name server on the device.

**rbridge-id** {*rbridge-id* | **all**}

If unspecified, executes only for current node. If a particular rbridge-id is specified then the command is executed for that node. If **rbridge-id all** is specified, the command executes for all nodes in the cluster.

## Modes

Privileged EXEC mode

## Usage Guidelines

The *trustpoint\_CAname* name needs to be the same as that of the trust point created by the **crypto ca trustpoint** command.

## Examples

Typical command example:

```
device# crypto ca enroll t1 country US state CA locality SJ organization BRC orgunit SFI common
myhost.extremenetworks.com protocol SCP host 10.70.12.102 user fvt directory /proj/crypto
Password: *****
```

## History

Release version	Command history
6.0.0	This command was introduced.

# crypto ca import

Imports the Identity Certificate for HTTPS security configuration.

## Syntax

```
crypto ca import { trustpointCA_name certificate directory remote_dir_name file cert_file host host_address protocol {FTP | SCP} user host_login password host_user_password [rbridge-id {rbridge-id | all}]}
```

```
no crypto ca import { trustpointCA_name [rbridge-id {rbridge-id | all}]}
```

## Parameters

*trustpointCA\_name*

Defines the name of the trust point you are authenticating. This name needs to be the same as that of the trust point created by the **crypto ca trustpoint** command. The string for the name can not be left blank. The length of the string can range from 1 through 64 characters.

**certificate directory** *remote\_dir\_name*

Defines the directory where the certification file resides.

**file** *cert\_file*

Defines the name of the certification file.

**host** *host\_address*

Defines the host name or IP address of the remote certificate server.

**protocol** {FTP | SCP}

Specifies the use of either FTP or SCP protocol for accessing the certification file.

**user** *host\_login*

Defines user name for the host server.

**password** *host\_user\_password*

Defines the password for the user name on the host server.

### NOTE

It is recommended to not list the password in command line for security purposes; the user will be prompted for the password.

**rbridge-id** {*rbridge-id* | **all**}

If unspecified, executes only for current node. If a particular rbridge-id is specified then the command is executed for that node. If **rbridge-id all** is specified, the command executes for all nodes in the cluster.

## Modes

Privileged EXEC mode

## Usage Guidelines

The *trustpoint\_CAname* name needs to be the same as that of the trust point created by the **crypto ca trustpoint** command.

Use the no form of the command to remove the Identity Certificate.

## Examples

Typical command example:

```
device# crypto ca import t1 certificate protocol SCP host 10.70.12.102 user fvt directory /users/crypto
file cacert.pem
Password: *****
```

## History

Release version	Command history
6.0.0	This command was introduced.

# crypto ca trustpoint

Defines the trust point for HTTPS security configuration.

## Syntax

`crypto ca trustpoint trustpointCA_name`

`no crypto ca trustpoint trustpointCA_name`

## Parameters

*trustpointCA\_name*

Defines the name of the trust point. The string for the name can not be left blank. The length of the string can range from 1 through 64 characters.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the `no crypto ca trustpoint` command to remove the trust point.

## Examples

Typical command example:

```
device(config-rbridge-id-1)# crypto ca trustpoint t1
```

Example using the no form of the command:

```
device(config-rbridge-id-1)# no crypto ca trustpoint t1
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# crypto key

Generates an RSA/ECDSA/DSA key pair to sign or encrypt and decrypt the security payload during security protocol exchanges for applications. You must sign and/or encrypt and decrypt the RSA/ECDSA/DSA key pair before you obtain a certificate for your device.

## Syntax

```
crypto key label key_label [rsa | ecdsa | dsa] [modulus key_size]
```

```
no crypto key label key_label
```

## Parameters

**label** *key\_label*

The name of the key pair.

**rsa**

Generates an RSA key pair.

**ecdsa**

Generates an ECDSA key pair.

**dsa**

Generates a DSA key pair.

**modulus** *key\_size*

Specifies the key size. The corresponding key sizes supported for each key type are:

- RSA: 1024 or 2048
- DSA: 1024
- ECDSA: 256,384, or 521

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the no form of this command to remove the key pair.

The key label must contain alphanumeric characters.

## Examples

Typical command example for generating the key pair.

```
device(config-rbridge-id-1)# crypto key label k1 rsa modulus 2048
device(config-rbridge-id-1)# do show running-config rbridge-id crypto
rbridge-id 1
crypto key label k1 rsa modulus 2048
```



The following is an example of using the no form of the command:

```
device(config-rbridge-id-1)# no crypto key label k1
```

## History

Release version	Command history
6.0.0	This command was introduced.

# custom-profile

Enables the user to specify a customized hardware profile.

## Syntax

```
custom-profile { kap profile_name } { bfd-l3 | bfd-vxlan | fcoe | lacp | rpvst | udd | xstp } [ hello-interval value | num-entry value ]
```

```
no custom-profile { kap profile_name } { bfd-l3 | bfd-vxlan | fcoe | lacp | rpvst | udd | xstp }
```

## Command Default

Custom profiles are disabled.

## Parameters

### kap

Enables the user to specify a customized Keep-Alive Protocol (KAP) profile.

### *profile\_name*

Name of the user-specified profile.

### bfd-l3

Configures protocol KAP parameters for BFD-L3 (Bidirectional Forwarding Detection for Layer 3).

### bfd-vxlan

Configures protocol KAP parameters for BFD VXLANs.

### fcoe

Configures protocol KAP parameters for FCoE.

### lacp

Configures protocol KAP parameters for Link Aggregation Control Protocol.

### rpvst

Configures protocol KAP parameters for Rapid Per-VLAN Spanning Tree.

### udd

Configures protocol KAP parameters for Unidirectional Link Detection.

### xstp

Configures protocol KAP parameters for any version of Spanning Tree Protocol.

### NOTE

Refer to the *Release Notes* for defaults and ranges for the above parameters.

## Modes

Hardware configuration mode.

## Usage Guidelines

### NOTE

The **hello-interval** and **num-entry** keywords are optional. If they are not specified, the default values are used.

Once a global custom KAP profile is defined, it can be applied to multiple switches in the cluster. The settings are dependent on the platform and the user application. For a custom profile to take effect, it has to be applied to a switch in RBridge ID configuration mode, by means of the **hardware-profile** command, and the switch must be rebooted. When a global custom profile is defined, only a generic validation process is performed, not a platform-specific validation.

Once a custom KAP profile is activated on one or more switches, the profile cannot be modified or deleted. To change custom profile settings on a switch, the user must first define a new custom KAP profile and apply it to the switch. Only when a custom profile is no longer applied to any switch in the cluster can that profile be deleted or modified, as shown in the examples below.

```
device(config-hardware)# no custom-profile kap myprofile
device(config-kap-myprofile)# no bfd-l3 hello-interval
```

## History

Release version	Command history
6.0.1	This command was introduced.

# dampening

Sets dampening parameters for the route in BGP address-family mode.

## Syntax

```
dampening { half-life reuse suppress max-suppress-time | route-map route-map }
no dampening
```

## Parameters

*half-life*

Number of minutes after which the route penalty becomes half its value. Range is from 1 through 45. Default is 15.

*reuse*

Minimum penalty below which the route becomes usable again. Range is from 1 through 20000. Default is 750.

*suppress*

Maximum penalty above which the route is suppressed by the device. Range is from 1 through 20000. Default is 2000.

*max-suppress-time*

Maximum number of minutes a route can be suppressed by the device. Default is 40.

*route-map*

Enables selection of dampening values established in a route map by means of the **route-map** command.

*route-map*

Name of the configured route map.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

Use the **no** form of this command to disable dampening.

Use **dampening** without operands to set default values for all dampening parameters.

To use the dampening values established in a route map, configure the route map first, and then enter the **route-map** command, followed by the name of the configured route map.

A full range of dampening values (*half-life, reuse, suppress, max-suppress-time*) can also be set by means of the **set as-path prepend** command.

## Examples

This example enables default dampening as an IPv4 address-family function.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# dampening
```

This example changes all the dampening values as an IPv6 address-family function.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# dampening 20 200 2500 40
```

This example applies the dampening half-life established in a route map and configures the route map using the **set dampening** command.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutemap permit 1
device(config-route-map-myroutemap/permit/1)# set dampening 20
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Usage guidelines were modified to indicate inheritance from the default VRF in support of Multi-VRF.

## Related Commands

[route-map](#), [set as-path](#), [set dampening](#)

# database-overflow-interval (OSPFv2)

Configures frequency for monitoring database overflow.

## Syntax

```
database-overflow-interval interval
no database-overflow-interval
```

## Command Default

0 seconds. If the device enters OverflowState, you must reboot before the device leaves this state.

## Parameters

*interval*

Time interval at which the device checks to see if the overflow condition has been eliminated. Valid values range from 0 through 86400 seconds.

## Modes

OSPF router configuration mode  
OSPF router VRF configuration mode

## Usage Guidelines

This command specifies how long a device that has entered the OverflowState waits before resuming normal operation of external LSAs. However, if the external link state database (LSDB) is still full, the device lapses back into OverflowState. If the configured value of the database overflow interval is zero, then the device never leaves the database overflow condition.

When the maximum size of the LSDB is reached (this is a configurable value in the *external-lsdb-limit* CLI), the device enters OverflowState. In this state, the device flushes all non-default AS-external-LSAs that the device had originated. The device also stops originating any non-default external LSAs. Non-default external LSAs are still accepted if there is space in the database after flushing. If no space exists, the Non-default external LSAs are dropped and not acknowledged.

For more information, refer to RFC 1765.

The **no** form of the command disables the overflow interval configuration.

## Examples

The following example configures a database-overflow interval of 60 seconds.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# database-overflow-interval 60
```

## Related Commands

[external-lsdb-limit \(OSPFv2\)](#)

# database-overflow-interval (OSPFv3)

Configures frequency for monitoring database overflow.

## Syntax

```
database-overflow-interval interval
no database-overflow-interval
```

## Command Default

0 seconds. If the router enters OverflowState, you must reboot before the router leaves this state.

## Parameters

*interval*

Time interval at which the device checks to see if the overflow condition has been eliminated. Valid values range from 0 through 86400 seconds (24 hours).

## Modes

OSPFv3 router configuration mode  
OSPFv3 router VRF configuration mode

## Usage Guidelines

This command specifies how long after a router that has entered the OverflowState before it can resume normal operation of external LSAs. However, if the external link state database (LSDB) is still full, the router lapses back into OverflowState.

When the maximum size of the LSDB is reached (this is a configurable value in the *external-lsdb-limit* CLI), the router enters OverflowState. In this state, the router flushes all non-default AS-external-LSAs that the router had originated. The router also stops originating any non-default external LSAs. Non-default external LSAs are still accepted if there is space in the database after flushing. If no space exists, the Non-default external LSAs are dropped and not acknowledged.

The **no** form of the command disables the overflow interval configuration.

## Examples

The following example configures a database-overflow interval of 120 seconds.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# database-overflow-interval 120
```



## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[external-lsdb-limit \(OSPFv3\)](#)

# debug access-list-log buffer

Configures or clears the ACL buffer.

## Syntax

```
debug access-list-log buffer { circular | linear } packet-count count-value
```

```
debug access-list-log buffer clear
```

```
no debug access-list-log buffer
```

## Parameters

**circular**

Specifies circular buffer type.

**linear**

Specifies linear buffer type.

**packet-count** *count-value*

Specifies a value from 64 through 2056.

**clear**

Clears the buffer contents.

## Modes

Privileged EXEC mode

## Usage Guidelines

D diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command to disable debugging.

## Examples

The following example clears the buffer.

```
device# debug access-list-log buffer clear
```

# debug bfd

Enables debugging for Bidirectional Forwarding Detection (BFD).

## Syntax

```
debug bfd { all | cli | error | event | ha | nsm | packet | rib | tmr { alert | critical | debug1 | debug2 | emergency | error | info |
  notify | warning } }
```

```
debug bfd dump { all | dest-ip | mac-table | nexthop-ip | prb-vrb | source-ip }
```

```
no debug bfd { all | cli | error | event | ha | nsm | packet | rib | tmr }
```

## Parameters

<b>all</b>	Debugs all BFD events.
<b>cli</b>	Debugs CLI BFD events.
<b>error</b>	Debugs information about possible BFD errors encountered.
<b>packet</b>	Debugs BFD control packets.
<b>rib</b>	Debugs the routing information base (RIB).
<b>tmr</b>	Debugs BFD timers.
<b>alert</b>	Debugs BFD alerts.
<b>critical</b>	Debugs BFD critical events.
<b>debug1</b>	Debugs BFD debug.
<b>debug2</b>	Debugs BFD verbose.
<b>emergency</b>	Debugs BFD emergencies.
<b>error</b>	Debugs BFD error messages.
<b>info</b>	Debugs general BFD information.
<b>notify</b>	Debugs BFD notifications.

**warning**

Debugs BFD warnings.

**dump**

Dumps BFD debug info to console.

**dest-ip**

Debugs destination IP address.

**mac-table**

Debugs MAC address table.

**nexthop-ip**

Debugs nexthop IP address.

**prb-vrb**

Displays ISL member port connecting the physical RB.

**source-ip**

Debugs source IP address.

## Modes

Privileged EXEC mode

## Usage Guidelines

D diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Enter **no debug bfd all** to reset debug levels to the default settings.

Enter **no debug bfd** followed by specific events or packet parameters to remove a specific BFD debugging configuration.

## Examples

This example sets debugging on notifications about BFD control packets.

```
device# debug bfd packet notify
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug dhcp packet buffer

Configures a buffer to capture DHCP packets.

## Syntax

```
debug dhcp packet buffer [ circular | linear ] [ packet-count 64-2056 ] [ vrf vrf-name ] [ interface <N> gigabitethernet
rbridge-id/slot/port ]
```

## Command Default

The buffer wraps around to overwrite earlier captures (circular).

## Parameters

### circular

Buffer wraps around to overwrite earlier captures.

### linear

Buffer stops capture when the packet-count value is reached.

### clear

Clears the packet buffer.

### all

Captures DHCP packets on all interfaces.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

This command configures the capturing buffer behavior by allowing captures to wrap and overwrite earlier captures or stop capturing when a packet-count limit is reached. The current buffer content is cleared when the configuration changes.

Use the **no** form of this command to disable debugging.

## Examples

The following example configures a buffer to capture 510 maximum packets in a circular fashion.

```
device# debug dhcp packet buffer circular packet-count 510
```

## History

Release version	Command history
7.0.0	This command was modified to include Usage Guidelines.

## Related Commands

[debug dhcp packet buffer interface](#), [debug dhcp packet buffer clear](#), [show debug dhcp packet buffer](#)

# debug dhcp packet buffer clear

Clears buffer content from DHCP packet capture.

## Syntax

```
debug dhcp packet buffer clear [ vrf vrf-name ]
```

## Parameters

**vrf** *vrf-name*

VRF name mapped to the VRF ID for which the buffer will be cleared. If this operand is not specified, the buffer for the default VRF ID is cleared.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Clears buffer content created from use of the **debug dhcp packet buffer interface** command to enable DHCP packet capture. If the DHCP packet capture is currently enabled, the buffer may fill again.

Use the **no** form of this command to disable debugging.

## Examples

The following example clears the buffer content of DHCP packets for the VRF titled "blue".

```
device# debug dhcp packet buffer clear blue
```

## History

Release version	Command history
7.0.0	This command was modified to include Usage Guidelines.

## Related Commands

[debug dhcp packet buffer interface](#)

# debug dhcp packet buffer interface

Enables and disables DHCP packet capture on a specific interface.

## Syntax

```
debug dhcp packet buffer interface [ <N>gigabitethernet rbridge-id/slot/port ] [ rx | tx ]
no debug dhcp packet buffer interface [ <N>gigabitethernet rbridge-id/slot/port [ rx | tx ]
debug dhcp packet buffer all
no debug dhcp packet buffer all
```

## Parameters

**all**

Enables DHCP packet capture on all switch interfaces.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**rx | tx**

Specifies whether to capture transmitted or received packets. If not specified, both are captured.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

The **all** operand replaces the *interface* operand.

Use the **no** form of this command to disable DHCP packet capture on a specific switch interface when used with **debug dhcp packet buffer interface** [*interface specifications*]

Use the **no** form of this command to disable the DHCP packet capture on all switch interfaces when used with **debug dhcp packet buffer all** .



You can specify a VLAN or physical port for capturing packets. If an interface is not specified, packets are captured on all interfaces.

## Examples

The following command enables DHCP packet capture for transmitting data on forty-gigabit Ethernet interface 1/0/1.

```
device# debug dhcp packet buffer interface fo 1/0/1 tx
```

The following command enables DHCP packet capture for receiving data on 10-gigabit Ethernet interface 1/0/1.

```
device# no debug dhcp packet buffer interface te 1/0/1 rx
```

The following command enables DHCP packet capture on all switch interfaces of switch 0.

```
device# debug dhcp packet buffer all
```

The following command disables DHCP packet capture on all switch interfaces of switch 0.

```
device# no debug dhcp packet buffer all
```

## History

Release version	Command history
7.0.0	This command was modified to include Usage Guidelines.

## Related Commands

[debug dhcp packet buffer clear](#), [debug dhcp packet buffer](#), [show debug dhcp packet](#)

# debug ip

Enables debugging for the IGMP and ICMP traffic on the switch.

## Syntax

```
debug ip packet [ interface interface-type interface-number [ vlan vlan_id ] | count { tx | rx } | icmp [ interface interface-type interface-number ] | count value | tx | rx | igmp [ interface interface-type interface-number ] | all | group multicast-grp-address ]
```

```
no debug ip packet
```

## Parameters

### packet

Enables IP packet debugging.

### interface

Displays the IP traffic for the specified interface only.

*interface-type*

Network interface type (external Ethernet interface, port-channel, or VLAN).

*interface-number*

Layer 2 or Layer 3 interface number.

### vlan *vlan\_id*

Specifies a VLAN.

### count *value*

Stops display after display count packets. Valid values range from 1 through 32256.

### tx

Counts only transmitted packets.

### rx

Counts only received packets.

### icmp

Displays the ICMP packets.

### igmp

Displays the IGMP packets.

### all

Enables all IGMP debugging.

### group

Enables IGMP debugging for multicast group.

*multicast-grp-address*

Multicast group address.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

When this feature is enabled, all IGMP or ICMP packets received or transmitted are displayed. Debugging can be enabled globally, per interface, or on a multicast group. Use the **no** form of this command to disable debugging.

## History

Release version	Command history
7.0.0	This command was modified to include Usage Guidelines.

# debug ip bgp

Displays information related to the processing of BGP4, with a variety of options.

## Syntax

```
debug ip bgp [ bfd | cli | dampening | events | general | graceful-restart | ip-prefix ip-addr/mask-len | ip-prefix-list name |
  keepalives | route-map name | route-selection | traces | updates [ rx | tx ] ] [ all-vrfs | vrf vrf-name ]
```

```
no debug ip bgp
```

## Parameters

### bfd

Displays information about BGP BFD.

### cli

Displays information about BGP CLI

### dampening

Displays BGP4 dampening.

### events

Displays all BGP4 events.

### general

Displays BGP4 common events.

### graceful-restart

Displays BGP graceful restart events.

### ip-prefix

Displays information filtered by IP prefix.

*ip-addr*

IPv4 address in dotted-decimal notation.

*mask-len*

IPv4 mask length in CIDR notation.

### ip-prefix-list

Displays information filtered by IP prefix list.

*name*

Name of IP prefix list.

### keepalives

Displays BGP4 keepalives.

### route-map

Displays configured route map tags.

*name*

Name of route map.

### route-selection

Displays BGP4 route selection.

**traces**

Displays BGP traces.

**updates**

Displays BGP4 updates.

**rx**

Displays BGP4 received updates.

**tx**

Displays BGP4 transmitted updates

**all-vrfs**

Specifies all VRFs.

**vrf**

Specifies a VRF instance or all VRFs.

*vrf-name*

Specifies a VRF instance

## Modes

Privileged EXEC mode

## Usage Guidelines

D diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command to disable debugging.

## Examples

The following example sets debugging on BGP4 events.

```
device# debug ip bgp events
```

The following example sets debugging on BGP4 graceful restart events.

```
device# debug ip bgp graceful-restart
```

The following example sets debugging on BGP4 events for VRF instance "red".

```
device# debug ip bgp events vrf red
```

## History

Release version	Command history
6.0.1	The <b>all-vrfs</b> and <b>vrf vrf-name</b> parameters were added to support Multi-VRF. The <b>bfd</b> parameter was added.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ip bgp neighbor

Displays information related to the processing of BGP4 for a specific neighbor.

## Syntax

```
debug ip bgp neighbor ip-addr [ all-vrfs | vrf vrf-name ]
no debug ip bgp neighbor ip-addr [ all-vrfs | vrf vrf-name ]
```

## Parameters

*ip-addr*  
IPv4 address in dotted-decimal notation.

**all-vrfs**  
Specifies all VRFs.

**vrf**  
Specifies a VRF instance or all VRFs.

*vrf-name*  
Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

The **no** form of the command disables debugging.

## History

Release version	Command history
6.0.1	The <b>all-vrfs</b> and <b>vrf vrf-name</b> parameters were added to support Multi-VRF.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ip fabric-virtual-gateway

Enables debugging of the Fabric-Virtual-Gateway protocol.

## Syntax

```
debug ip fabric-virtual-gateway { all | fabric | nsm | cli | garp | interface ve vlan-id | ipv4 | ipv6 }
```

```
no debug ip fabric-virtual-gateway {all | fabric | nsm | cli | garp | interface ve vlan-id | ipv4 | ipv6 }
```

## Command Default

None

## Parameters

### all

Debugs all that follows the debug option.

### fabric

Debugs fabric related events like fabric up or down.

### nsm

Debug NSM related events line interface up or down or add or delete.

### cli

Debugs CLI related executions or arguments.

### garp

Debugs sent gratuitous ARP.

### interface ve *vlan-id*

Debugs session level events specified by VLAN ID.

### *ipv4*

Debugs all IPv4 session events.

### *ipv6*

Debugs all IPv6 session events.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Enter the **no** form of the command to disable debugging for the Fabric-Virtual-Gateway protocol.

## Examples

The following example enables debugging for all Fabric-Virtual-Gateway protocols.

```
device# debug ip fabric-virtual-gateway all
```

The following example enables debugging of the fabric for the Fabric-Virtual-Gateway protocol.

```
device# debug ip fabric-virtual-gateway fabric
```

The following example enables debugging of interface VE 2000 for the Fabric-Virtual-Gateway protocol.

```
device# debug ip fabric-virtual-gateway interface ve 2000
```

## History

Release version	Command history
5.0.1	This command was introduced.
7.0.0	This command was modified to include updated Usage Guidelines.



# debug ip igmp

Enables or disables debugging for IGMP information.

## Syntax

```
debug ip igmp { all | errors | group A.B.C.D | packet | rx | tx | interface { <N>gigabitethernet rbridge-id/slot/port | port-channel
  number | vlan vlan_id } }
```

```
no debug ip igmp
```

## Parameters

### all

Enables all debugs.

### errors

Enables only error type debugs, such as memory allocation failures etc.

### group *A.B.C.D*

Specifies the group address, as a subnet number in dotted decimal format (for example, 10.0.0.1), as the allowable range of addresses included in the multicast group.

### packet

Enables debug for query/reports per the chosen option.

### rx

Specifies only ingressing flow debugs to be captured in traces.

### tx

Specifies only egressing packet flows to be captured in traces.

### vlan

Specifies the VLAN to be monitored.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### port-channel *number*

Specifies the interface is a port-channel. The number of available channels range from 1 through 6144.

### vlan *vlan\_id*

Specifies which VLAN interface to display the snooping configuration related information. Refer to the Usage Guidelines.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

When debugging is enabled, all of the IGMP packets received and sent and IGMP-host related events are displayed.

Use the **no** form of this command to disable debugging.

## Examples

The following example displays specific debugs enabled on RBridge 1 and 5.

```
device# show debug ip igmp

IGMP debugging status: rbridge_id 1
-----
errors          : off
group           : off
packets         : on
query           : on
report          : on
direction       : none
vlan            : none
l2_port         : none

IGMP debugging status: rbridge_id 5
-----
errors          : off
group           : off
packets         : on
query           : on
report          : on
direction       : both
vlan            : te0/35
l2_port         : none
```

## History

Release version	Command history
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ip ospf

Enables debugging for the IP Open Shortest Path First (OSPF) protocol.

## Syntax

```
debug ip ospf { adj | all-vrfs | bfd | dev | error | events | flood | log-debug-message | log-empty-lsa | ls-id A.B.C.D | lsa-
generation | max-metric | neighbor A.B.C.D | packet | retransmission | route A.B.C.D | spf | vrf name }
```

```
no debug ip ospf
```

## Command Default

IP OPSF debugging is disabled.

## Parameters

<b>adj</b>	Adjacency related debugs.
<b>all-vrfs</b>	Information for all VRFs instances in a cluster.
<b>bfd</b>	Information for OSPF BFD.
<b>dev</b>	Developer debug options.
<b>error</b>	Displays possible errors encountered during time.
<b>events</b>	Events-related debugs.
<b>flood</b>	Flooding-related debugs.
<b>log-debug-message</b>	Debugs message logging.
<b>log-empty-lsa</b>	Empties LSA logging.
<b>ls-id A.B.C.D</b>	Link state ID (LSID) debugging for the link-state ID that you specify.
<b>lsa-generation</b>	LSA generation-related debugging.
<b>max-metric</b>	Stub Router Advertisement.
<b>neighbor A.B.C.D</b>	Neighbor debugging for the neighbor that you specify.

debug ip ospf

**packet**

Packet debugs.

**retransmission**

Retransmission events.

**route** *A.B.C.D*

Route debugs for the router that you specify.

**spf**

SPF trace.

**vrf** *name*

Debug information for VRF.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Enter **no debug ip ospf** to disable IP OPSF debugging.

## Examples

This example enables adjacency-related debugs.

```
device# debug ip ospf adj
```

## History

Release version	Command history
6.0.1	The <b>bfd</b> parameter was added.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ip pim

Enables debugging for IP Protocol Independent Multicast.

## Syntax

```
debug ip pim { add-del-oif | bootstrap | group | join-prune | nbr-change | packets | parent | regproc | route-change | rp |  
source | state | all }
```

```
no debug ip pim all
```

## Command Default

All flags are disabled.

## Parameters

### add-del-oif

Controls the OIF change flag.

### bootstrap

Controls the bootstrap processing flag.

### group

Controls the processing for a group flag.

### join-prune

Controls the Join/Prune processing flag.

### nbr-change

Controls the neighbor changes flag.

### packets

Controls the packet processing flag.

### parent

Controls the parent change processing flag.

### regproc

Controls the register processing flag.

### route-change

Controls the route changes flag.

### rp

Controls the Rendezvous Point (RP) processing flag.

### source

Controls the processing for a source flag.

### state

Controls the state processing flag.

### all

Controls all of the states.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no debug ip pim all** command to disable debugging.

## History

Release version	Command history
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ip rtm

Enables debugging for IP RTM.

## Syntax

```
debug ip rtm { A.B.C.D | all | counters { clear | show } dump | errors | fib-comm | nexthop | port | vrf }
```

## Command Default

IP RTM debugging is disabled.

## Parameters

*A.B.C.D*

Debugs the route specified by this IP address.

**all**

Enables all debugs.

**counters**

Enables debug counters.

**clear**

Clears debug counters.

**show**

Shows debug counters.

**dump**

Shows database dump.

**errors**

Enables internal error debugs.

**fib-comm**

Debugs communications between the forwarding information base and the routing table manager.

**nexthop**

Enables next-hop debugs.

**port**

Enables port database debugs.

**vrf**

Enables VRF debugs.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

To debug the route specified by the IP address 192.145.12.1:

```
device# debug ip rtm 192.145.12.1
```

To show a database dump:

```
device# debug ip rtm dump
```

```
Interface      IP-Address      OK?  Method Status      Protocol VRF
Gi 190/0/1     0xbe2a640c      YES  manual up        up        default-vrf
Ve 128         0xa52a800c      YES  manual admin/down up        default-vrf
Ve 1001        0x0a010101      YES  manual admin/down up        default-vrf
Ve 1001        0x65010101      YES  manual admin/down up        default-vrf
Lo 1           0xa02a0c0c      YES  manual up        up        default-vrf
mgmt 1         0x0a14eabe      YES  manual up        up        default-vrf
IP Static Routing Table - 1 entries:
addr: 0x1021f4b8, top 0x1021f590, count 1, default 0 ffffffff
      Type 2
Route pool:
pool: 101e3bd0, unit_size: 32, initial_number:128, upper_limit:2000000000
      total_number:128, allocated_number:1, alloc_failure 0
      flag: 0, pool_index:1, avail_data:102207b8
Route Entry Pool:
pool: 101e3c80, unit_size: 432, initial_number:128, upper_limit:2000000000
      total_number:128, allocated_number:1, alloc_failure 0
      flag: 0, pool_index:1, avail_data:10221950
NextHop Settings
Update: no, Update-always no, Update-Timer 0 Check-Nexthops no
Recur: yes, Levels 3, Default-enable no
vrf-count 0, vrf-resolved yes
Protocols: < connected>
Nexthops List
 [7] 0xa14e801 hash 7 paths 1 upd last-update-time 0 -> 0xa14e801 mgmt 1
NextHop List End
```

## History

Release version	Command history
7.0.0	This command was modified to include Usage Guidelines.



# debug ip vrf

Displays information related to VRF.

## Syntax

```
debug ip vrf ip-addr
```

```
no debug ip vrf
```

## Parameters

*ip-addr*

IPv4 address in dotted-decimal notation.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command to disable debugging.

## Examples

To display VRF-related information:

```
device# debug ip vrf
```

## History

Release version	Command history
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ipv6 bgp

Displays debug information related to BGP processing for IPv6 prefix lists.

## Syntax

```
debug ipv6 bgp ipv6-prefix ipv6-address /mask [ all-vrfs | vrf vrf-name ]
debug ipv6 bgp ipv6-prefix name [ all-vrfs | vrf vrf-name ]
debug ipv6 bgp ipv6-prefix-list name [ all-vrfs | vrf vrf-name ]
no debug ipv6 bgp ipv6-prefix ipv6-address /mask [ all-vrfs | vrf vrf-name ]
no debug ipv6 bgp ipv6-prefix name [ all-vrfs | vrf vrf-name ]
no debug ipv6 bgp ipv6-prefix-list name [ all-vrfs | vrf vrf-name ]
```

## Parameters

*ipv6-address /mask*  
Specifies an IPv6 address and network mask.

**all-vrfs**  
Specifies all VRFs.

**vrf**  
Specifies a VRF instance or all VRFs.

*vrf-name*  
Specifies a VRF instance

*name*  
Specifies a prefix list name.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command to disable debugging.

## Examples

This example enables debugging for IPv6 prefix list "myv6list" for VRF instance "red".

```
device# debug ipv6 bgp ipv6-prefix-list myv6list vrf red
```

This example enables debugging for a specified IPv6 address for all VRFs.

```
device# debug ipv6 bgp ipv6-prefix 2001::/16 all-vrfs
```

## History

Release version	Command history
6.0.1	The <b>all-vrfs</b> and <b>vrf vrf-name</b> parameters were added to support Multi-VRF.
7.0.0	The Usage Guidelines were updated.

# debug ipv6 bgp neighbor

Displays debug information related to BGP processing for a specified neighbor.

## Syntax

```
debug ipv6 bgp neighbor ipv6-addr [ all-vrfs | vrf vrf-name ]  
no debug ipv6 bgp neighbor ipv6-addr [ all-vrfs | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor.

**all-vrfs**

Specifies all VRFs.

**vrf**

Specifies a VRF instance or all VRFs.

*vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

The **no** form of the command disables debugging.

## Examples

The following example sets debugging for a neighbor.

```
device# debug ipv6 bgp neighbor 2000::1
```

The following example specifies that BGP keepalives for a specified neighbor appear in debugging messages.

```
device# debug ip bgp keepalive  
device# debug ipv6 bgp neighbor 2001::1
```

The following example sets debugging for a neighbor for VRF instance "red".

```
device# debug ipv6 bgp neighbor 2000::1 vrf red
```

The following example sets debugging for a neighbor for all VRFs.

```
device# debug ipv6 bgp neighbor 2000::1 all-vrfs
```

## History

Release version	Command history
6.0.1	The <b>all-vrfs</b> and <b>vrf vrf-name</b> parameters were added to support Multi-VRF.

# debug ipv6 dhcpv6 packet buffer

Enables IPv6 DHCPv6 packet capture on an interface or all interfaces.

## Syntax

```
debug ipv6 dhcpv6 packet buffer { all | [ interface [<N> gigabitethernet rbridge-id/slot/port| ve vlan_id ] [ rx | tx ] }
no debug ipv6 dhcpv6 packet buffer { all | interface [<N> gigabitethernet rbridge-id/slot/port| ve vlan_id ] [ rx | tx ] }
```

## Command Default

None

## Parameters

**all**

Specifies all buffer packets.

**<N> gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**rx**

Specifies the receive direction.

**tx**

Specifies the receive direction.

**ve***vlan\_id*

Specifies a virtual Ethernet interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command with the appropriate keywords to disable packet capture.

## Examples

To enable ICMPv6 packet capture on all interfaces:

```
device# debug ipv6 dhcpv6 packet buffer all
```

To disable Neighbor Discovery packet capture on an interface in the transmit direction:

```
device# no debug ipv6 dhcpv6 packet buffer int te 54/0/22 tx
```

## History

Release version	Command history
5.0.1	This command was introduced.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ipv6 mld

Displays information related to IPv6 Multicast Listener Discovery (MLD), with a variety of options.

## Syntax

```
debug ipv6 mld { all | errors } | group | interface [ vlanvlan_id ] l2-port [ <N> gigabitethernet | port-channel ] | packet [ query | report ] | rx | tx }
```

```
no debug ipv6 mld { all | errors } | group | interface [ vlanvlan_id ] l2-port [ <N> gigabitethernet | port-channel ] | packet [ query | report ] | rx | tx }
```

## Command Default

None

## Parameters

### all

Displays all information.

### errors

Displays error conditions.

### group

Displays information for a match group.

### interface

Displays information for a specified interface.

### vlanvlan\_id

Specifies a VLAN. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

### l2-port

Displays information for a physical or LAG port.

### N gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port).

### port-channel

Specifies a port-channel interface.

### packet

Displays information related to MLD packets.

### query

Displays information for MLD query packets.

### report

Displays information for MLD report packets.



- rx** Displays information for incoming MLD packets.
- tx** Displays information for outgoing MLD packets.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command to disable debugging.

## Examples

To view all IPv6 MLD information:

```
device# debug ipv6 mld all
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ipv6 nd

Displays information related to IPv6 Neighbor Discovery (ND)

## Syntax

`debug ipv6 nd`  
`no debug ipv6 nd`

## Command Default

None

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command to disable debugging.

## Examples

To display information related to IPv6 ND:

```
device# debug ipv6 nd
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ipv6 ospf

Enables debugging for the IPv6 Open Shortest Path First (OSPF) protocol.

## Syntax

```
debug ipv6 ospf { all-vrfs | bfd | gr-helper | ipsec | ism | ism-events | ism-status | lsa | lsa-flooding | lsa-generation | lsa-install | lsa-inter-area | lsa-maxage | lsa-refresh | match-prefix | nsm | nsm-events | nsm-status | packet | packet-dd | packet-hello | packet-lsa-ack | packet-lsa-req | packet-lsa-update | route | route-calc-external | route-calc-inter-area | route-calc-intra-area | route-calc-spf | route-calc-transit | route-install | virtual-link [ all-vrfs | rbridge-id { rbridge-id } | vrf vrfname ] }
```

```
no debug ip ospf
```

## Command Default

IPv6 OPSF debugging is disabled.

## Parameters

### all-vrfs

Information for all VRFs instances in a cluster.

### bfd

Information for OSPF BFD.

### gr-helper

Information for the OSPFv3 graceful restart helper.

### ipsec

Information for IPsec events.

### ism

Interface State Machine.

### ism-events

Interface State Machine events.

### ism-status

Interface State Machine status.

### lsa

Link State Advertisements.

### lsa-flooding

Link State Advertisements flooding.

### lsa-generation

Link State Advertisements generation.

### lsa-install

Link State Advertisements install.

### lsa-inter-area

Inter-area Link State Advertisements.

**lsa-maxage**

Link State Advertisements max aging.

**lsa-refresh**

Link State Advertisements refreshing.

**match-prefix**

Enables match prefix in debug output.

**nsm**

Neighbor state machine.

**nsm-events**

Neighbor state machine events.

**nsm-status**

Neighbor state machine status.

**packet**

OSPFv3 packets.

**packet-dd**

OSPFv3 data description packets.

**packet-hello**

OSPFv3 hello packets.

**packet-lsa-ack**

OSPFv3 LSA ack packets.

**packet-lsa-req**

OSPFv3 LSA Request packets.

**packet-lsa-update**

OSPFv3 LSA Update packets.

**route**

OSPFv3 routes.

**route-calc-external**

OSPFv3 external route calculation.

**route-calc-inter-area**

OSPFv3 inter area route calculation.

**route-calc-intra-area**

OSPFv3 intra area route calculation.

**route-calc-spf**

OSPFv3 spf route calculation.

**route-calc-transit**

OSPFv3 transit route calculation.

**route-install**

OSPFv3 route install.

**virtual-link**

OSPFv3 virtual links.

**vrf name**

Debug information for VRF.

**rbridge-id** *rbridge-id*

The physical, loopback, and SVI interfaces specific to the selected RBridge.

*ipv6-addr* Specifies an IPv6 address in dotted-decimal notation.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Enter **no debug ip ospf** to disable IP OPSF debugging.

## Examples

To enable OSPFv3 graceful restart helper debugs:

```
device# debug ipv6 ospf gr-helper
OSPFv3: gr-helper debugging is on
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>bfd</b> parameter was added.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug ipv6 packet buffer

Enables IPv6 packet capture on an interface or all interfaces.

## Syntax

```
debug ipv6 packet buffer { all | circular packet-count count | clear | interface [<N> gigabitethernet rbridge-id/slot/port] ve
  vlan_id] [ rx | tx ] | linear packet-count count }
```

```
no debug ipv6 packet buffer { all | circular packet-count count | clear | interface [<N> gigabitethernet rbridge-id/slot/port] ve
  vlan_id] [ rx | tx ] | linear packet-count count }
```

## Command Default

None

## Parameters

### all

Specifies all buffer packets.

### circular packet-count

Specifies the number of packets in a circular buffer. Range is from 64 through 2056.

### clear

Clears contents of the buffer.

### *<N>* gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *<N>* **gigabitethernet** with the desired operand (for example, **ten** **gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

#### rx

Specifies the receive direction.

#### tx

Specifies the receive direction.

### ve *vlan\_id*

Specifies a virtual Ethernet interface.

### linear packet-count

Specifies the number of packets in a linear buffer. Range is from 64 through 2056.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostics commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Use the **no** form of this command with the appropriate keywords to disable packet capture.

## Examples

To enable ICMPv6 packet capture on all interfaces:

```
device# debug ipv6 packet buffer all
```

To disable Neighbor Discovery packet capture on an interface in the transmit direction:

```
device# no debug ipv6 packet buffer int te 54/0/22 tx
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.0.0	This command was modified to include updated Usage Guidelines.

# debug lacp

Enables or disables debugging for the Link Aggregation Control Protocol (LACP).

## Syntax

```
debug lacp { all | cli | event | ha | pdu [ rx { all | interface <N>gigabitethernet rbridge-id/slot/port | tx { all | sync | timer | trace
level number } }
```

```
no debug lacp
```

## Command Default

LACP debugging is disabled.

## Parameters

**all**

Turns on all debugging.

**cli**

Turns on command line interface debugging.

**event**

Turns on event debugging.

**ha**

Echo HA events to the console.

**pdu**

Echo PDU content to the console.

**rx all**

Turns on debugging for received LACP packets on all interfaces.

**rx interface**

Turns on debugging for received LACP packets on the specified interface.

**interface**

Specifies the interface to be monitored.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.



**tx all**

Turns on debugging for transmitted LACP packets on all interfaces.

**tx interface**

Turns on debugging for transmitted LACP packets on the specified interface.

**sync**

Echo synchronization to consoles.

**timer**

Echo timer expiration to console.

**trace level *number***

Specifies the trace level number. Valid values range from 1 through 7.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Enter **terminal monitor** to display debugging outputs on a particular cmsh session.

Enter **no debug lacp** to disable LACP debugging.

## Examples

To enable debugging of LACP PDUs for transmitted and received packets on all interfaces:

```
device# debug lacp pdu tx all
device # debug lacp pdu rx all

device# show debug lacp
LACP rx debugging is on
LACP tx debugging is on
```

## History

Release version	Command history
7.0.0	This command was modified to include updated Usage Guidelines.

## Related Commands

[show debug lacp](#)

# debug lldp dump

Dumps debugging information for the Link Layer Discovery Protocol (LLDP) to the console.

## Syntax

```
debug lldp dump { all | [ <N> gigabitethernet rbridge-id/slot/port ] [ both ] } [ detail [ both | rx | tx ] }
```

## Command Default

LLDP debugging is disabled.

## Parameters

### all

Dumps all information to the console.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

### both

Turns on debugging for both transmit and receive packets.

### detail

Turns on debugging with detailed information.

### both

Turns on detailed debugging for both transmit and receive packets.

### rx

Turns on detailed debugging for only received LLDP packets.

### tx

Turns on detailed debugging for only transmitted LLDP packets.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

To dump LLDP informaton:

```
device# debug lldp dump all
LLDP Interface Debug Information for Fo 2/0/49
Admin Status:  RX_TX
Associated Profile:
Link-level FCoE Priority: 0x08 (Configured: No)
Link-level iSCSI Priority: 0x10 (Configured: No)
Link Properties:
  CEE Incapable
  FCoE LLS not Ready
  FCF-Forward Disabled
Sending TLVs:
  CHASSIS_ID: 0x50eb1a173ff1 (MAC)
  PORT_ID: Fo 2/0/49 (IF Name)
  TTL: Hold (4) x Interval (30)
  SYSTEM_NAME
  IEEE_DCBX
  DCBX_FCOE_APP
  DCBX_FCOE_LOGICAL_LINK
  Configured FCoE App
  Configured FCoE Link
  DCBX_CTRL
<truncated>
```

## History

Release version	Command history
7.0.0	This command was modified to include Usage Guidelines.

## Related Commands

[show debug lldp](#)

# debug lldp packet

Enables or disables debugging for the Link Layer Discovery Protocol (LLDP).

## Syntax

```
debug lldp packet { all | [ <N>gigabitethernet rbridge-id/slot/port ] [ both ] } [ detail [ both | rx | tx ] ]
no debug lldp packet { all | interface <N>gigabitethernet rbridge-id/slot/port }
```

## Command Default

LLDP debugging is disabled.

## Parameters

### all

Turns on LLDP packet debugging on all interfaces.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### both

Turns on debugging for both transmit and receive packets.

### detail

Turns on debugging with detailed information.

### both

Turns on detailed debugging for both transmit and receive packets.

### rx

Turns on detailed debugging for only received LLDP packets.

### tx

Turns on detailed debugging for only transmitted LLDP packets.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Enter **terminal monitor** to display debugging outputs on a particular cmsh session.

Enter **no debug lldp packet** to disable LLDP debugging.

## Examples

To enable debugging of LLDP for both received and transmitted packets on the 10-gigabit Ethernet interface 0/1:

```
device# debug lldp packet interface tengigabitethernet 0/1 both
```

```
device# show debug lldp
```

```
LLDP debugging status:
Interface te0/1      : Transmit Receive
```

## History

Release version	Command history
7.0.0	This command was modified to include Usage Guidelines

## Related Commands

[show debug lldp](#)

# debug show bp-stats interface

Displays buffer-pool queue (BPQ) information, BPQ mapping, and the total and in-use resources for each queue.

## Syntax

```
debug show bp-stats interface <N>gigabitethernet rbridge-id / slot / port
```

## Parameters

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following example displays BPQ information for a 10-Gb Ethernet interface.

```
switch# debug show bp-stats interface tengigabitethernet 1/0/1
```

	Rx/Trap	HwDrop	Tx	Queue Depth
BPQ-0	0	0	0	5a
BPQ-1	0	0	0	5a
BPQ-2	0	0	0	5a
BPQ-3	0	0	0	5a

## History

Release version	Command history
7.0.0	This command was introduced.

# debug show qos drop-reason interface

Displays QoS dropped packets—with the drop reason—for one or all interfaces.

## Syntax

```
debug show qos drop-reason interface { all | <N>gigabitethernet rbridge-id / slot / port }
```

## Parameters

**all**

Displays drop-reason information for all interfaces (in a chassis that spans multiple RBridges).

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following example displays QoS drop-reason information for forty-gigabit Ethernet interface 1/0/52.

```
device# debug show qos drop-reason interface Fo 1/0/52

Only non-zero counters are shown
Drop Reason Statistics for Interface Fo 1/0/52
=====
                L2C_ACL_DROP: 0000004a5e
                L2C_NXT_HOP_ERR: 00000000fa
```



The following example displays QoS drop-reason information for ten-gigabit Ethernet interface 60/0/50.

```
device# debug show qos drop-reason interface Te 60/0/50

Only non-zero counters are shown
Drop Reason Statistics for Interface Te 60/0/50:3
=====
                L3C_DROP_BY_RTE_DROP: 000040525a
                L2C_ACL_ACL_DROP: 0000016434
                L2C_ACL_DROP: 0000043503
                L2C_NXT_HOP_ERR: 0000012ca7
TX Source Supression DROP :0000000f2d
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.0.0	Additional drop categories were added.

# debug spanning-tree

Enables debugging for the Spanning Tree Protocol (STP).

## Syntax

```
debug spanning-tree { all | bpdu [ rx | tx [ all | interface port-channel number | <N>gigabitethernet slot/port ] ] }
no debug spanning-tree { all | bpdu [ rx | tx [ all | interface port-channel number | <N>gigabitethernet slot/port ] ] }
```

## Command Default

STP debugging is disabled.

## Parameters

### all

Turns on spanning tree packet debugging on all interfaces.

### bpdu

Turns on Bridge Protocol Data Unit debugging.

### rx

Turns on debugging for only received spanning-tree packets.

### tx

Turns on debugging for only transmitted spanning-tree packets.

### interface

Specifies the interface to be monitored.

### port-channel *number*

Specifies the port-channel interface. Valid values range from 1 through 6144.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

Enter **terminal monitor** to display debugging outputs.

Enter **no debug spanning-tree** to disable debugging.

## Examples

To enable debugging of spanning-tree for both Rx and Tx on the 10-gigabit Ethernet interface 0/1:

```
device# debug spanning-tree bpdu rx interface tengigabitethernet 0/1
```

```
device# debug spanning-tree bpdu tx interface tengigabitethernet 0/1
```

```
device# show debug spanning-tree
```

```
MSTP debugging status:
Spanning-tree rx debugging is off
Te 0/1 rx is on
Spanning-tree tx debugging is off
Te 0/1 tx is on
```

## History

Release version	Command history
7.0.0	This command was modified to include updated Usage Guidelines.

## Related Commands

[show debug spanning-tree](#)

# debug udd packet

Enables debugging for the Unidirectional Link Detection (UDLD) protocol.

## Syntax

```
debug udd packet [ all | { interface [ <N>gigabitethernet rbridge-id/slot/port ] } { both | rx | tx }
no debug udd packet
```

## Command Default

UDLD debugging is disabled.

## Parameters

**all**

Activates UDLD debugging on all ports on the switch.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tenGigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot/port*

Specifies a valid slot and port number.

**both**

Sets debugging for both received and transmitted packets.

**rx**

Sets debugging for received packets only.

**tx**

Sets debugging for transmitted packets only.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

When debugging is enabled UDLD PDUs are written to the console as they are transmitted and/or received on one or all ports.

Use the **show debug udd** command to view your current debug settings.

Use the **no** form of this command to turn off either all dumping of UDLD PDUs or dumping on a specific port.

## Examples

To turn on debugging of transmitted packets on a specific tengigabitethernet interface:

```
device# debug udd packet interface tengigabitethernet 5/0/1 tx
```

## History

Release version	Command history
7.0.0	This command was modified to include updated Usage Guidelines.

## Related Commands

[show debug udd](#)

# debug vrrp

Enables debugging for the Virtual Router Redundancy Protocol (VRRP).

## Syntax

**debug vrrp all**

**debug vrrp events**

**debug vrrp packets** { **interface** { *<N>gigabitethernet rbridge-id/slot/port* | **ve** *vlan\_id* } | **recv** | **sent** }

**debug vrrp session** *VRID*

**no debug vrrp all**

**no debug vrrp events**

**no debug vrrp packets** { **interface** { *<N>gigabitethernet rbridge-id/slot/port* | **ve** *vlan\_id* } | **recv** | **sent** }

**no debug vrrp session** *VRID*

## Parameters

**all**

Debugs all VRRP events, packets, and sessions.

**events**

Debugs all VRRP events.

**packets interface**

Debugs packets for an interface that you specify. Also enables the *recv* and *sent* parameters.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**ve** *vlan\_id*

Specifies the VLAN number for a virtual Ethernet (VE) interface.

**packets recv**

Debugs packets received.

**packets sent**

Debugs packets sent.

**session** *VRID*

Specifies the virtual group ID to debug. Valid values range from 1 through 128.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

When debugging is enabled, event and packet information for all virtual groups or for a specific interface are captured..

This command is for VRRP and VRRP-E. VRRP-E supports only the VE interface type.

Enter **no debug vrrp all** with to disable all VRRP debugging.

Enter **no debug vrrp** followed by specific events or packet parameters to remove a specific VRRP debugging configuration.

## Examples

To set debugging on sent and received packets for a 10-gigabit Ethernet interface that has an *rbridge-id/slot/port* of 121/0/50:

```
device# debug vrrp packets interface tengigabitethernet 121/0/50
```

To set debugging for a session for a VRRP virtual group called *vrrp-group-25* :

```
device# debug vrrp session 25
```

## History

Release version	Command history
7.0.0	This command was modified to include updated Usage Guidelines.

## Related Commands

[show debug vrrp](#)

# default-behavior

Configures default table-miss behavior for an OpenFlow logical instance.

## Syntax

```
default-behavior { drop | send-to-controller }
```

## Command Default

See the Usage Guidelines.

## Parameters

### drop

Drops packets in case of a table miss.

### send-to-controller

Sends packets to the controller in case of a table miss.

## Modes

OpenFlow logical instance configuration mode

## Usage Guidelines

By default, the device drops packets that do not match any of the programmed flows (a table miss). Use this command to configure the device to forward the packets to the controller instead of dropping them.

## Examples

The following example configures default table-miss behavior for an OpenFlow logical instance to send packets to the controller.

```
device(config)# rbridge-id 12
device(config)rbridge-id-12)# openflow logical-instance 1
device(config-logical-instance-1)# default-behavior send-to-controller
```

The following example configures default table-miss behavior for an OpenFlow logical instance to drop the packets.

```
device(config)# rbridge-id 12
device(config)rbridge-id-12)# openflow logical-instance 1
device(config-logical-instance-1)# default-behavior drop
```

## History

Release version	Command history
6.0.1	This command was introduced.



# default-config enable

Allows the switch to always reboot with its default configuration and rejoin the cluster after a reboot. The device obtains its configuration from the principal node. Enabling this feature solves most node-segmentation issues.

## Syntax

```
default-config enable
no default-config enable
```

## Command Default

This feature is disabled by default.

## Modes

RBridge ID configuration mode.

## Usage Guidelines

Use the **no** form of this command to disable this feature on a switch.

Extreme recommends not enabling this feature on *every* node (especially core nodes) in the cluster.

This feature cannot be enabled on the principal node, nor can principal priority be set on a node which has this feature enabled.

## Examples

The following example enables the default configuration feature on a device whose RBridge ID is 10:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# default-config enable
```

## History

Release version	Command history
6.0.0	This command was introduced.

# default-information-originate (BGP)

Configures the device to originate and advertise a default BGP route.

## Syntax

**default-information-originate**

**no default-information-originate**

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the default.

## Examples

The following example originates and advertises a default BGP4 route.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# default-information-originate
```

The following example originates and advertises a default BGP4+ route for VRF "red".

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# default-information-originate
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# default-information-originate (OSPFv2)

Controls distribution of default information to an OSPFv2 device.

## Syntax

```
default-information-originate [ always ] [ metric metric ] [ metric-type { type1 | type2 } ] [ route-map name ]
no default-information-originate
```

## Command Default

The default route is not advertised into the OSPFv2 domain.

## Parameters

### always

Always advertises the default route. If the route table manager does not have a default route, the router advertises the route as pointing to itself.

### metric *metric*

specifies the cost for reaching the rest of the world through this route. If you omit this parameter and do not specify a value using the *default-metric* router configuration command, a default metric value of 1 is used. Valid values range from 1 through 65535. The default is 10.

### metric-type

Specifies how the cost of a neighbor metric is determined. The default is **type1**. However, this default can be changed with the **metric-type** command.

#### type1

Type 1 external route.

#### type2

Type 2 external route.

### route-map *name*

Specifies that the default route is generated if the route map is satisfied. This parameter overrides other options. If the **set metric** and **set metric-type** commands are specified in the route-map, the command-line values of metric and metric-type if specified, are "ignored" for clarification.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

This configuration provides criteria for the redistribution of any default routes found in the route table manager (RTM), whether static or learned from another protocol, to its neighbors.

The corresponding route-map should be created before configuring the **route-map** option, along with the **default-information-originate** command. If the corresponding route-map is not created beforehand, an error message is displayed stating that the route-map must be created.

The route-map option cannot be used with a non-default address in the match conditions. The default route LSA is not generated if a default route is not present in the routing table and a **match ip address** condition for an existing non-default route is configured in the route-map. The **match ip address** command in the route-map is a no-op operation for the default information originate command.

The **no** form of the command disables default route origination.

## Examples

The following example creates and advertises a default route with a metric of 30 and a type 1 external route.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# default-information-originate metric 30 metric-type type1
```

## Related Commands

[default-metric \(OSPF\)](#), [metric-type](#)

# default-information-originate (OSPFv3)

Controls distribution of default information to an OSPFv3 device.

## Syntax

```
default-information-originate [ always ] [ metric metric ] [ metric-type { type1 | type2 } ]
no default-information-originate
```

## Command Default

The default route is not advertised into the OSPFv3 domain.

## Parameters

### always

Always advertises the default route. If the route table manager (RTM) does not have a default route, the router advertises the route as pointing to itself.

### metric *metric*

Used for generating the default route, this parameter specifies the cost for reaching the rest of the world through this route. If you omit this parameter, the value of the **default-metric** command is used for the route. Valid values range from 1 through 65535.

### metric-type

Specifies the external link type associated with the default route advertised into the OSPF routing domain.

#### type1

The metric of a neighbor is the cost between itself and the router plus the cost of using this router for routing to the rest of the world.

The default is **type1**.

#### type2

The metric of a neighbor is the total cost from the redistributing routing to the rest of the world.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

This configuration provides criteria for the redistribution of any default routes found in the RTM (whether static or learned from another protocol) to its neighbors.

The **no** form of the command disables default route origination.

## Examples

The following example specifies a metric of 20 for the default route redistributed into the OSPFv3 routing domain and an external metric type of Type 2.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# default-information-originate metric 20 metric-type
type2
```

# default-local-preference

Enables setting of a local preference value to indicate a degree of preference for a route relative to that of other routes.

## Syntax

```
default-local-preference num  
no default-local-preference
```

## Parameters

*num*

Local preference value. Range is from 0 through 65535. The default is 100.

## Modes

BGP configuration mode

## Usage Guidelines

Local preference indicates a degree of preference for a route relative to that of other routes. BGP4 neighbors can send the local preference value as an attribute of a route in an UPDATE message.

## Examples

The following example sets the local preference value to 200.

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# router bgp  
device(config-bgp-router)# default-local-preference 200
```

# default-metric (BGP)

Changes the default metric used for redistribution.

## Syntax

**default-metric** *value*

**no default-metric**

## Command Default

The default metric value is 1.

## Parameters

*value*

Metric value. Range is from 0 through 4294967295.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the default.

## Examples

The following example changes the default metric used for redistribution to 100.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# default-metric 100
```

The following example changes the default metric used for redistribution to 200 for VRF "green".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf green
device(config-bgp-ipv6u-vrf)# default-metric 200
```



## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# default-metric (OSPF)

Sets the default metric value for the OSPFv2 or OSPFv3 routing protocol.

## Syntax

```
default-metric metric
no default-metric
```

## Parameters

*metric*  
OSPF routing protocol metric value. Valid values range from 1 through 65535. The default is 10.

## Modes

- OSPF router configuration mode
- OSPFv3 router configuration mode
- OSPF router VRF configuration mode
- OSPFv3 router VRF configuration mode

## Usage Guidelines

This command overwrites any incompatible metrics that may exist when OSPFv2 or OSPFv3 redistributes routes. Therefore, setting the default metric ensures that neighbors will use correct cost and router computation.

The **no** form of the command restores the default setting.

## Examples

The following example sets the default metric to 20 for OSPF.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# default-metric 20
```

## History

Release version	Command history
5.0.0	Support was added for OSPFv3.

## Related Commands

[default-information-originate \(OSPFv2\)](#)

# default-passive-interface

Marks all OSPFv2 and OSPFv3 interfaces passive by default.

## Syntax

**default-passive-interface**

**no default-passive-interface**

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

When you configure the interfaces as passive, the interfaces drop all the OSPFv2 and OSPFv3 control packets.

You can use the **ip ospf active** and **ip ospf passive** commands in interface subconfiguration mode to change active/passive state on specific OSPFv2 interfaces. You can use the **ipv6 ospf active** and **ipv6 ospf passive** commands in interface subconfiguration mode to change the active and passive state on specific OSPFv3 interfaces.

The **no** form of the command disables the passive state.

## Examples

The following example marks all OSPFv2 interfaces as passive for a specified RBridge.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router ospf
device(config-router-ospf-vrf-default-vrf)# default-passive-interface
```

## History

Release version	Command history
5.0.0	Support was added for OSPFv3.

# delay

For an implementation of an event-handler profile, specifies a delay from when a trigger is received until execution of the event-handler action.

## Syntax

**delay** *seconds*

**no delay**

## Command Default

There is no delay from when a trigger is received until execution of the event-handler action.

## Parameters

*seconds*

Specifies the number of seconds from when a trigger is received until the execution of the specified action begins. Valid values are 0 or a positive integer.

## Modes

Event-handler activation mode

## Usage Guidelines

The **no** form of this command resets the **delay** setting to the default 0 seconds.

## Examples

The following example specifies a delay of 60 seconds.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# delay 60
```

The following example resets **delay** to the default value of 0 seconds.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# no delay
```

## History

Release version	Command history
6.0.1	This command was introduced.

# delete

Deletes a user-generated file from the flash memory.

## Syntax

**delete** *file*

## Parameters

*file*

The name of the file to be deleted.

## Modes

Privileged EXEC mode

## Usage Guidelines

The delete operation is final; there is no mechanism to restore the file.

This command is supported only on the local switch.

System configuration files cannot be deleted. If you try to delete a system configuration file, an appropriate message is displayed.

## Examples

To delete a user-generated copy of a configuration file:

```
device# dir
total 24
drwxr-xr-x  2 root    sys      4096 Feb 13 00:39 .
drwxr-xr-x  3 root    root     4096 Jan  1 1970 ..
-rwxr-xr-x  1 root    sys       417 Oct 12 2010 myconfig
-rwxr-xr-x  1 root    sys       417 Oct 12 2010 defaultconfig.novcs
-rwxr-xr-x  1 root    sys       697 Oct 12 2010 defaultconfig.vcs
-rw-r--r--  1 root    root     6800 Feb 13 00:37 startup-config
device# delete myconfig

% Warning: File will be deleted (from flash:)!
Continue?(y/n): y
```

## Related Commands

[copy](#), [dir](#), [rename](#), [show file](#)

# description (event-handler)

Defines a description for an event-handler profile.

## Syntax

**description** *description-text*

**no description**

## Command Default

No description is defined.

## Parameters

*description-text*

Characters describing the event-handler profile. The string can be 1 through 128 ASCII characters in length. Do not use the ? character. If you need to use ! or \, precede each with \.

## Modes

Event-handler configuration mode

## Usage Guidelines

An event-handler profile supports only one description.

To delete a description, use the **no** form of this command.

To change a description, you do not need to first delete the existing description. Just create a new description.

## Examples

The following example defines a description for eventHandler1.

```
device# configure terminal
device(config)# event-handler eventHandler1
device(config-event-handler-eventHandler1)# description This is a sample description.
```

## History

Release version	Command history
7.0.0	This command was introduced.

# description (interfaces)

Specify a string that contains the description of a specified interface..

## Syntax

**description** *line*

**no description**

## Parameters

*line*

Specifies characters describing the interface. The string must be between 1 and 63 ASCII characters in length.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no description** to remove the interface description.

## Examples

To set the string describing internal 10-gigabit Ethernet interface 101/0/1:

```
device# configure terminal
device(config)# interface tengigabitethernet 101/0/1
device(conf-if-te-101/0/1)# description converged_101
```

# description (LLDP)

Specifies a string that contains the LLDP description.

## Syntax

**description** *string*

**no description**

## Parameters

*string*

Characters describing LLDP. The string must be between 1 and 50 ASCII characters in length.

## Modes

Protocol LLDP and profile configuration modes

## Usage Guidelines

Enter **no description** to remove the LLDP description.

The LLDP description can also be configured for a specific LLDP profile. When you apply an LLDP profile on an interface using the **lldp profile** command, it overrides the global configuration. If a profile is not present, then the default global profile is used until you create a valid profile.

## Examples

To set the strings describing LLDP:

```
device(conf-lldp)# description Extreme-LLDP
```

To set the strings describing LLDP for a specific LLDP profile, test2, enter the following:

```
device(conf-lldp)# profile test1
device(config-profile-test1)# description test2
device(config-profile-test1)#
```



# description (Port Mirroring)

Specifies a string that contains the description of the Port Mirroring session.

## Syntax

**description** *line*

**no description**

## Parameters

*line*

Specifies string that contains the description of the Port Mirroring session. The string must be between 1 and 64 ASCII characters in length.

## Modes

Monitor session configuration mode

## Usage Guidelines

The string displayed in the running-config file to describe the Port Mirroring session.

Enter **no description** to remove the port mirroring description.

## Examples

To set the string describing monitor session 1:

```
device# configure terminal
device(config)# monitor session 1
device(config-mon-sess-1)# description server group 1 switch-cmsh
```

## Related Commands

[monitor session](#)

# description (VRRP)

Describes a Virtual Router Redundancy Protocol extended (VRRP-E) interface.

## Syntax

**description** *description*

**no description**

## Parameters

*description*

Characters describing the VRRP-E interface. The string must be between 1 and 64 ASCII characters in length.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

Enter **no description** to remove the description.

## Examples

To describe the VRRP-E group 10 interface:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# interface ve 25
device(config-ve-25)# vrrp-extended-group 10
device(config-vrrp-extended-group-10)# description vrrpe_group_10
```

## Related Commands

[vrrp-group](#), [vrrp-extended-group](#)

# destination

Designates the destination interface for the snooping data for flow-based SPAN.

## Syntax

```
destination dest_ifname
```

```
no destination dest_ifname
```

## Parameters

*dest\_ifname*

The name of the destination interface.

## Modes

Monitor session mode

## Usage Guidelines

Use the **no destination** *dest\_ifname* command to delete the destination interface.

## Related Commands

[span session](#)

# device-connectivity

Designates a port as being connected to a storage device.

## Syntax

```
device-connectivity { DAS | NAS | none | iSCSI }
```

## Parameters

### DAS

Indicates that the interface's device connectivity to a DAS storage device.

### NAS

Indicates that the interface's device connectivity to a NAS storage device.

### none

Indicates that the port is not connected to a storage device.

### iSCSI

Indicates that the interface's device connectivity to iSCSI storage device.

## Modes

Interface configuration mode

## Usage Guidelines

If a port is configured as being connected to iSCSI storage, then the Monitoring and Alerting Policy Suite (MAPS) feature uses this field to monitor the port against thresholds for IP storage port group.

The **show running-config interface** command displays the status for a specific port.

## Examples

Typical command example designating a tengigabit Ethernet port as as connectivity for an iSCSI device.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/2/5
device(conf-if-te-1/2/5)# device-connectivity iscsi
```

## History

Release version	Command history
6.0.1	This command was introduced.

# dhcp auto-deployment enable

Enables DHCP Automatic-Deployment (DAD) on the switch.

## Syntax

```
dhcp auto-deployment enable
```

## Command Default

DAD is disabled by default.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command causes a cold/disruptive reboot and requires that Telnet, secure Telnet, or SSH sessions be restarted.

Scenario 1: When you enable DAD and the system starts to reboot, the DAD process is triggered after configuration replay is complete.

For dual management module (MM) chassis (VDX 8770 chassis), the DAD process waits for the dual MM to be in sync before starting the requested firmware download. However, if you manually issue **firmwaredownload -sb** during this period (after DAD is triggered and before MM is in sync), DAD fails because the previous firmware download takes precedence. If you manually issue **firmwaredownload -sb** before DAD is triggered, DAD will fail for the same reason.

Scenario 2: You issue the command to enable DAD (answer "Yes" when prompted), but before the system reboot, there is an HA failover. DAD will be cancelled. You must enable DAD from the new active switch.

Scenario 3: You issue the command to enable DAD, but after the system reboot is invoked, takeover occurs (the previous standby switch becomes the new active switch), DAD will proceed.

Scenario 4: You manually issue the **firmwaredownload** command, but before the firmware download is completed, you enable DAD from the CLI and answer "Yes" when prompted to reboot the switch. When the switch boots up, even if the DAD process detects that the firmware download is needed, it will fail during the sanity check because the previous incomplete firmware download takes precedence. DAD fails.

Scenario 5: If you do not power down the secondary node when running DAD on the principal node, the following outcomes are observed:

- The DAD principal node and secondary node form the cluster, but without the principal role. DAD fails on the principal node.
- The DAD principal node and secondary node form the cluster with the principal role. DAD proceeds on the principal node because the secondary node does not affect DAD.
- The DAD principal node and secondary node do not form the cluster. DAD proceeds on the principal node because the secondary node does not affect DAD.

dhcp auto-deployment enable

## Examples

The following example shows enabling DAD:

```
device# dhcp auto-deployment enable
```

## Related Commands

[show dadstatus](#)

# dhcpd enable

Enables the DHCP server functionality on the VDX 6740T device.

## Syntax

`dhcpd enable`

`no dhcpd enable`

## Modes

Rbridge ID configuration mode

## Usage Guidelines

If DHCP is enabled on the management interface, you must disable DHCP and configure a compatible static IP address before you enable the DHCP server. To make the configuration effective you must perform a manual reboot. If DHCP fails to start, an error report is displayed with a raslog.

The device does not reboot when DHCP server is disabled. DHCP relay will be enabled when the device reboots.

## Examples

The following example enables DHCP server on the VDX 6740T device.

```
device# configure terminal
device(config)# rbridge 1
device(config-rbridge-id-1)# dhcpd enable
Please reboot the system to make configuration effective
device(config-rbridge-id-1)#
```

## History

Release version	Command history
7.1.0	This command was introduced.

# dhcpc restart

Restarts the dhcpc server.

## Syntax

```
dhcpc restart
```

## Modes

Privileged EXEC mode

## Usage Guidelines

If the DHCP server is enabled and running, you can update the dhcpc.conf file using copy scp://<username>:<password>@hostname/<filepath> flash://dhcpc.conf. You can then restart the dhcpc server with the new dhcpc.conf.

## Examples

The following example restarts the DHCP server.

```
device# dhcpc restart
```

## History

Release version	Command history
7.1.0	This command was introduced.



# diag burninerrclear

Clears the error logs that are stored in the nonvolatile memory. These error logs are stored during POST and systemVerification failures. Error logs are automatically cleared during system verification.

## Syntax

```
diag burninerrclear
```

## Modes

Privileged EXEC mode

## Examples

Typical output for this command.

```
device# diag burninerrclear

Clearing errLog for slot M2
Clearing errLog for slot S1
Clearing errLog for slot S2
Clearing errLog for slot S3
Clearing errLog for slot L4
```

## Related Commands

[diag clearerror](#), [show diag burninerrshow](#), [show diag burninstatus](#)

# diag clearerror

Clears the diagnostic errors encountered during offline diagnostic tests.

## Syntax

`diag clearerror`

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is valid only on fixed-configuration switches.

## Examples

To clear the diagnostic failure status:

```
device# diag clearerror
```

## Related Commands

[diag burninerrclear](#), [show diag burninerrshow](#), [show diag burninstatus](#)

# diag dport-test interface

Allows the user to start, stop, and restart diagnostic testing on a port enabled as a D\_Port (diagnostic port), as well as to specify the frame size and number of frames to be transmitted in testing.

## Syntax

```
diag dport-test interface { <N>gigabitethernet<bridge-id/slot/port> { restart | start | stop } [ setarg { framesize frame_size |
nframe number_of_frames }
```

## Command Default

See Parameters for the default testing conditions.

## Parameters

**<N>gigabitethernet**

Specifies a TenGigabitEthernet, FortyGigabitEthernet, or HundredGigabitEthernet interface.

**restart**

Restarts D\_Port testing.

**start**

Starts D\_Port testing.

**stop**

Stops D\_Port testing.

**setarg**

Specifies frame size, number of frames, or both.

**framesize** *frame\_size*

Specifies the size of frames, in bytes, to be transmitted in testing. The range is from 36 bytes through 2112 bytes. The default is 1024.

**nframe** *number\_of\_frames*

Specifies the number of frames, in millions (1,000,000), to be transmitted in testing. The range is from 1 through 1024 million. The default is 1 million.

## Modes

Privileged EXEC mode

## Usage Guidelines

There is no **no** form of this command.

Dynamic D\_Port functionality is enabled by default on TenGigabitEthernet, FortyGigabitEthernet, and HundredGigabitEthernet ports. If the default functionality is set to none, it must be reenabled by means of either the **fabric dport mode dynamic** or **fabric dport mode static** commands.

Only the **start** keyword supports the **setarg** keyword. When the **setarg** keyword is specified, at least one of the following keywords must be specified: **framesize**, **nframe**.

## Examples

To start D\_Port testing on a TenGigabitEthernet port and set the number of frames to 2,000,000 and the frame size to 1800 bytes.

```
device# diag dport-test interface Tengigabitethernet 51/0/13 start nframe 2 framesize 1800
```

```
=====
Interface          Status
=====
Te 51/0/13         Test started. To view results run "show dport-test interface Te 51/0/13"
=====
```

### NOTE

As noted in the test response, you can use the **show dport-test interface** command to view the results.

## History

Release version	Command history
7.0.0	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

# diag portledtest

Runs various action modes on the port LED tests and validates the functionality on a given slot-based switch or fixed-configuration switch.

## Syntax

```
diag portledtest [ action pattern ] [ ethernet rbridgeid/slot/port ] [ fibrechannel rbridgeid/slot/port ] [ npass count ] [ slot slot_id ]
```

## Command Default

All the ports are tested in a switch.

The default number of times to perform the test is 1.

The default **action** is cycle\_all

## Parameters

### **action** *pattern*

Specifies the LED pattern. Action choices are as follows:

#### *blink-amber*

Blink Port status LED amber

#### *blink-green*

Blink Port status LED green

#### *cycle-all*

Cycle all Port LEDs

#### *status-amber*

Turn Port status LED amber

#### *status-green*

Turn Port status LED green

#### *turn-off*

Turn Port status LED off

### **ethernet**

The logical Ethernet interface name, which is mutually exclusive from the Fibre Channel parameter. By default, all ports are tested.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number

**fibrenchannel**

The logical Fibre Channel interface name, which is mutually exclusive from the Ethernet parameter. By default, all ports are tested.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**npass count**

Specifies the number of times to perform this test. Valid values range from 1 through 10. The default value is 1.

**slot slot\_id**

Specifies the slot identifiers for slot-based systems only.

## Modes

Privileged EXEC mode (with the chassis disabled in offline mode)

## Usage Guidelines

This test can be run on a single port or on all ports in the blade (slot-based switches) or the switch (fixed-configuration switches).

During abnormal termination, the system might be in unusable state. Perform reload to reboot the chassis or switch to recover.

In slot-based systems, the blade under test undergoes a reset and/or a reinitialization sequence as part of cleanup.

The *rbridge-id* is an optional parameter. If the *rbridge-id* is not specified, the test is assigned to the local RBridge ID.

**CAUTION**

This is a disruptive command. You must disable the switch and chassis before running the test. In addition, you must reload or fastboot the switch or chassis after the test has completed running.

## Examples

The following commands allow you to run various action modes on the LEDs and validate the functionality.

In slot-based switches:

```
device# diag portledtest action cycle-all slot L1

% Info: This test should be run to completion. Please do not abort while it is executing.
Running portledtest...
Turning Port Status LEDs OFF...
Turning Port Status LEDs AMBER...
Turning Port Status LEDs GREEN...
Turning Port Status LEDs BLINK GREEN...
Turning Port Status LEDs BLINK AMBER...
portLedTest on slot L1 PASSED
% Info: Resetting the blade. Please wait till it gets initialized...
device#
```

In fixed-configuration switches:

```
device# diag portledtest
```

```
% Info: This test should be run to completion. Please do not abort while it is executing.
Running portledtest ...
Testing Ethernet ports..
STATUS LED OFF test
STATUS LED GREEN test
STATUS LED AMBER test
STATUS LED BLINK GREEN test
STATUS LED BLINK AMBER test
Testing FC ports..
STATUS LED OFF test
STATUS LED GREEN test
STATUS LED AMBER test
STATUS LED BLINK GREEN test
STATUS LED BLINK AMBER test
PASSED.
```

## Related Commands

[diag portloopbacktest](#), [diag post enable](#), [diag turboramtest](#)

# diag portloopbacktest

Runs the port loopback test on a given slot-based switch or fixed-configuration switch. You can run this test on a single port or on all ports in the blade (slot-based switches) or switch (fixed-configuration switches). This functional test verifies the ability of each port to transmit and receive frames by setting up the loopback at various levels and speed modes.

## Syntax

```
diag portloopbacktest [ ethernet rbridgeid/slot/port ] [ fibrechannel rbridgeid/slot/port ] [ lbmode loopback_mode ] [ nframes
  count ] [ slot slot_id ] [ spdmode mode ]
```

## Command Default

Number of frames (**nframes**) is 16.

Loopback mode (**lbmode**) is 2.

Speed mode (**spdmode**) depends on the platform. On a 10 Gbps port, the default speed mode is 10.

## Parameters

### ethernet

The logical Ethernet interface name, which is mutually exclusive from the Fibre Channel parameter. By default, all ports are tested.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number

### fibrechannel

The logical Fibre Channel interface name, which is mutually exclusive from the Ethernet parameter. By default, all ports are tested.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### lbmode *mode*

Specifies the loopback point for the test. Valid values are 1 (external) or 2 (internal). The default is 2.

### nframes *count*

Specifies the number of frames to send. Valid values range from 1 through 16. The default is 16.

### slot *slot\_id*

Specifies the slot identifiers for slot-based systems only.



**spdmode mode**

Specifies the speed mode for the test. This parameter controls the speed at which each port operates during the test. Valid parameters are as follows: 1 Gbps 2 Gbps 4 Gbps 8 Gbps 10 Gbps 16 Gbps 40 Gbps 100 Gbps

## Modes

Privileged EXEC mode (with the chassis disabled, in offline mode)

## Usage Guidelines

During abnormal termination, the system might be in unusable state. Perform reload to reboot the chassis or switch to recover.

In slot-based systems, the blade under test undergoes a reset or a reinitialization sequence as part of cleanup.

The *rbridge-id* is an optional parameter. If the *rbridge-id* is not specified, the test is assigned to the local RBridge ID.

**CAUTION**

**This is a disruptive command. You must disable the chassis and switch before running the test. In addition, you must reload or fastboot the switch or chassis after the test has completed running.**

## Examples

In slot-based switches:

```
device# diag portloopbacktest slot S1

% Info: This test should be run to completion. Please do not abort while it is executing.
Running portloopbacktest...
<..cut..>
portLoopbackTest on ports 0-143 PASSED
portLoopbackTest on slot S1 PASSED
% Info: Resetting the blade. Please wait till it gets initialized...
```

In fixed-configuration switches:

```
device# diag portloopbacktest

% Info: This test should be run to completion. Please do not abort while it is executing.
Running portloopbacktest .....
PASSED.
```

## Related Commands

[diag portledtest](#), [diag post enable](#), [diag turboramtest](#)

# diag post enable

Enables and disables the power-on self-test (POST).

## Syntax

```
diag post [ rbridge-id ] enable
```

```
no diag post [ rbridge-id ] enable
```

## Command Default

POST is enabled.

## Parameters

*rbridge-id*

Specifies an RBridge ID on which POST is run.

**enable**

Enables the power-on self-test on the specified switch.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no diag post [ *rbridge-id* ] enable** to disable the POST for that RBridge.

## Examples

The following example enables the POST for a RBridge: .

```
device# configure terminal
Entering configuration mode terminal
device(config)# diag post rbridge-id 1 enable
```

The following example disables the POST for an RBridge ID:

```
device# configure terminal
device(config)# no diag post rbridge-id 1 enable
```

## Related Commands

[show running-config diag post](#)

# diag prbstest

Runs the Pseudo Random Bit Sequence (PRBS) test on a given slot to verify the back end connections between the line card (LC) and switch fabric module (SFM).

This command also verifies the internal blade connections when executed in LC.

## Syntax

```
diag prbstest slot { L1 | L2 | S1 | S2 ... } pattern { pattern }
```

## Command Default

The default PRBS pattern is PRBS7.

## Parameters

**slot** *slot*

Specifies the slot ID, from 1 through 6. This test is applicable for slot-based systems only.

**pattern** *pattern*

Specifies the PRBS pattern, from 1 through 8. Valid values are **PRBS7** , **PRBS23** , and **PRBS31** . The default is PRBS7, which is the least stressful pattern, whereas PRBS31 is the most stressful pattern.

## Modes

Privileged EXEC mode (with the chassis disabled, in offline mode)

## Usage Guidelines

This test is not supported on fixed-configuration switches, nor can it be run on a per-port basis

During abnormal termination, the system might be in unusable state. Perform reload to reboot the chassis or switch to recover.

In slot-based systems, the blade under test undergoes a reset and/or a reinitialization sequence as part of the cleanup process.



### CAUTION

This is a disruptive command. You must disable the chassis and switch before running the test. In addition, you must reload or fastboot the switch or chassis after the test has completed running.

## Examples

In slot-based switches:

```
device# diag prbstest slot L6 pattern PRBS7

% Info: This test should be run to completion. Please do not abort while it is executing.
Running prbstest...
Initializing ASICs & Ports...
Performing Link Training from L6 to S1
Performing Link Training from L6 to S2
Performing Link Training from L6 to S3
Performing Link Training from L6 to S4
Performing Link Training from L6 to S5
<..cut..>
slot S6 ASIC 1 Port 15 Tap0: 0x08 Tap1: 0x33 Tap2: 0x20
Performing Link Testing from L6 to S1
Performing Link Testing from L6 to S2
Performing Link Testing from L6 to S3
Performing Link Testing from L6 to S4
Performing Link Testing from L6 to S5
Performing Link Testing from L6 to S6
prbsTest on slot L6 PASSED
```

## Related Commands

[diag portledtest](#), [diag portloopbacktest](#), [diag turboramtest](#)

# diag setcycle

Configures the user-defined parameters required for the system verification test.

## Syntax

```
diag setcycle
```

## Command Default

Referto the Usage Guidelines.

## Modes

Privileged EXEC mode

## Usage Guidelines

If, after you enter the **diag setcycle** command, you respond with **yes** , the following settings are the default values:

- *num\_of\_runs* : 1. Valid values for number of runs are 1 through 25.
- *min\_lb\_mode* : 2. Valid values for minimum loopback mode are 1 (external) or 2 (internal). If set to 1, all the external user ports must be connected with small form-factor pluggable devices (SFPs) and loopback plugs.
- *pled\_passes* : 1. Valid values for the number of portLedTest loops are 1 through 10.
- *tbr\_passes* : 1. Valid values for the number of turboRamTest loops are 1 through 10. This parameter is not supported on fixed configuration switches.
- *plb\_nframes* :16. Valid values for the number of portLoopbackTests are 4 through 16.

If you respond with **no** , the system prompts you for these values.

## Examples

To change the value of num\_of\_runs parameter to 3:

```
device# diag setcycle num_of_runs 3
Setting number_of_runs to 3.
Committing changes to configuration
```

In slot-based switches:

```
0 is not a valid number of passes. See sample below.
ronteel28# diag setcycle num_of_runs 0
```

```
-----^
syntax error: "0" is out of range.
device# diag setcycle
Do you want use default values [Y/N]?      : y

DEFAULT - KEYWORD      : COMMENT
replacing 2 with default 1
  1 - number_of_runs    : number of passes of verify
  2 - min_lb_mode       : Limits -lb_mode of tests
VERIFY - label : Label for run start and stop messages
  1 - tbr_passes        : turboramtest number of passes
replacing 8 with default 16
  16 - plb_nframes      : portloopbacktest number of frames default speed
  1 - pled_passes       : portledtest number of passes
  1 - prbs_p7           : LC Backplane test with pattern PRBS7+
  16 - cplb_nframes     : portloopbacktest in Core Blade number of frames
Committing changes to configuration
```

```
device# diag setcycle
```

```
Do you want use default values [Y/N]?      : y
```

```
DEFAULT - KEYWORD      : COMMENT
  1 - number_of_runs    : number of passes of verify (0=infinite)
  2 - min_lb_mode       : Limits -lb_mode of tests
  0 - sof               : Enable stop testing on first fail
VERIFY - label : Label for run start and stop messages
  1 - tbr_passes        : turboramtest number of passes
  16 - plb_nframes      : portloopbacktest number of frames default speed
  0 - plb5_nframes      : portloopbacktest (lb_mode 5) number of frames default speed
  0 - plb7_nframes      : portloopbacktest (lb_mode 7) number of frames
  0 - pled_action       : portledtest action for glowing all led's
  1 - pled_passes       : portledtest number of passes
  1 - prbs_p7           : LC Backplane test with pattern PRBS7+
  0 - prbs_p23          : LC Backplane test with pattern PRBS23+
  0 - prbs_p31          : LC Backplane test with pattern PRBS31+
  0 - cprbs_p7          : SFM Backplane test with pattern PRBS7+
  0 - cprbs_p23         : SFM Backplane test with pattern PRBS23+
  0 - cprbs_p31        : SFM Backplane test with pattern PRBS31+
  16 - cplb_nframes     : portloopbacktest in Core Blade number of frames
  0 - cplb7_nframes    : portloopbacktest in Core Blades (lb_mode 7) number of frames
```

In fixed-configuration switches:

```
device# diag setcycle
```

```
Do you want use default values [Y/N]? : y
```

```
DEFAULT - KEYWORD : COMMENT
replacing 3 with default 1
1 - number_of_runs : number of passes of verify (0=infinite)
2 - min_lb_mode    : Limits -lb_mode of tests
1 - tbr_passes     : turboramtest number of passes
16 - plb_nframes   : portloopbacktest number of frames default speed
Committing changes to configuration
```

## Related Commands

[show diag setcycle](#)

# diag systemverification

Runs a combination of various hardware diagnostic tests based on the parameters set using the diag setcycle command.

## Syntax

```
diag systemverification [ short ] [ stop ]
```

## Command Default

If *short* is not specified, all the burn-in parameters that control the number of frames are run.

## Parameters

### short

Sets the burn-in parameters that control the number of frames to one for a quick run.

### stop

Stops the current systemVerification run.

## Modes

Privileged EXEC mode (with the chassis disabled in offline mode)

## Usage Guidelines

The primary use for this command is software regression testing, or a quick validation that all hardware is operational.



### CAUTION

This is a disruptive command. You must disable the chassis and switch before running the test. In addition, you must reload or fastboot the switch or chassis after the test has completed running.

Error logs are cleared automatically during system verification.

To check the current run status, enter the **show diag burninstatus** command.

All errors are stored in the non-volatile memory. You can check the error status using the **show diag burninerrshow** command.

During abnormal termination or when terminated by using the stop parameter, the system might be in unusable state. Perform a reload to reboot the chassis or switch to recover.

In slot-based systems, the blade under test undergoes a reset and/or a reinitialization sequence as part of the cleanup process.

## Examples

To run various tests, such as the memory and portloopback tests, with various combinations:

```
device# diag systemverification

% Info: This test should be run to completion. Please do not abort while it is executing.
systemverification: burnin parameters.
CURRENT - KEYWORD : DEFAULT
1 - number_of_runs : 1
2 - min_lb_mode : 2
1 - tbr_passes : 1
16 - plb_nframes : 16
<..cut..>
```



# diag turboramtest

This test performs a series of low-level structural tests to determine the basic health of the PCI or PCIe bus and the memories inside the switch ASIC.

## Syntax

```
diag turboramtest [ passcnt count ] [ slot slot_id ]
```

## Command Default

The pass count (**passcnt**) is 1.

## Parameters

**passcnt** *count*

Specifies the number of test repetitions. By default, the test runs once. Valid values range from 1 through 10.

**slot** *slot\_id*

Specifies the slot ID. This is mandatory for slot-based systems only.

## Modes

Privileged EXEC mode (with the chassis disabled in offline mode).

## Usage Guidelines

During abnormal termination, the system might be in unusable state. Perform reload to reboot the chassis or switch to recover.

In slot-based systems, the blade under test undergoes a reset and/or a reinitialization sequence as part of the cleanup process.



### CAUTION

This is a disruptive command. You must disable the chassis and switch before running the test. In addition, you must reload or fastboot the switch or chassis after the test has completed running.

## Examples

In slot-based switches:

```
device# diag turboramtest slot S2

% Info: This test should be run to completion. Please do not abort while it is executing.
Running turboramtest...
Initializing ASIC 0 for BIST
Initializing ASIC 1 for BIST
Initializing ASIC 2 for BIST
turboRamTest on ASIC 0 PASSED
turboRamTest on ASIC 1 PASSED
turboRamTest on ASIC 2 PASSED
turboRamTest on slot S2 PASSED
% Info: Resetting the blade. Please wait till it gets initialized...
completed.
```

diag turboramtest

In fixed-configuration switches:

```
device# diag turboramtest
```

```
% Info: This test should be run to completion. Please do not abort while it is executing.  
Running turboramtest .....  
PASSED.
```

# dir

Lists the contents of the device flash memory.

## Syntax

dir

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

## Examples

The following example lists the contents of the flash memory.

```
device# dir

total 24
drwxr-xr-x  2 root    sys      4096 Feb 13 00:39 .
drwxr-xr-x  3 root    root     4096 Jan  1 1970 ..
-rwxr-xr-x  1 root    sys       417 Oct 12 2010 myconfig.vcs
-rwxr-xr-x  1 root    sys       417 Oct 12 2010 defaultconfig.novcs
-rwxr-xr-x  1 root    sys       697 Oct 12 2010 defaultconfig.vcs
-rw-r--r--  1 root    root     6800 Feb 13 00:37 startup-config
```

## Related Commands

[copy](#), [delete](#), [rename](#), [show file](#)

# disable (Fabric-Virtual-Gateway)

Disables the Fabric-Virtual-Gateway session on the VE interface.

## Syntax

**disable**  
**no disable**

## Command Default

None

## Modes

Fabric-Virtual-Gateway on an RBridge VE interface IPv4 or IPv6 configuration mode

## Usage Guidelines

The **no** form of the command inherits the interface VE state from the global configuration.

The session can be disabled at the RBridge level even if it is enabled at the global level.

## Examples

The following example shows how to disable a session on a VE interface.

```
device(config)# rbridge-id 55
device(config-rbridge-id-55)# interface ve 2000
device(config-rbridge-Ve-2000)# ipv6 fabric-virtual-gateway
device(config-ipv6-fabric-virtual-gw)# disable
```

## History

Release version	Command history
5.0.1	This command was introduced.

# distance (BGP)

Changes the default administrative distances for eBGP, iBGP, and local BGP.

## Syntax

**distance** *external-distance internal-distance local-distance*

**no distance**

## Parameters

*external-distance*

eBGP distance. Range is from 1 through 255.

*internal-distance*

iBGP distance. Range is from 1 through 255.

*local-distance*

Local BGP4 and BGP4+ distance. Range is from 1 through 255.

## Modes

BGP configuration mode

## Usage Guidelines

To select one route over another according to the source of the route information, the device can use the administrative distances assigned to the sources. The administrative distance is a protocol-independent metric that IP devices use to compare routes from different sources. Lower administrative distances are preferred over higher ones.

## Examples

The following example configures the device to change the administrative distance.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# distance 100 150 200
```

# distance (OSPF)

Configures an administrative distance value for OSPFv2 and OSPFv3 routes.

## Syntax

```
distance { external | inter-area | intra-area } distance  
no distance
```

## Command Default

The administrative distance value for OSPFv2 and OSPFv3 routes is 110.

## Parameters

### external

Sets the distance for routes learned by redistribution from other routing domains.

### inter-area

Sets the distance for all routes from one area to another area.

### intra-area

Sets the distance for all routes within an area.

### *distance*

Administrative distance value assigned to OSPF routes. Valid values range from 1 through 255. The default is 110.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

You can configure a unique administrative distance for each type of OSPF route.

The distances you specify influence the choice of routes when the device has multiple routes from different protocols for the same network. The device prefers the route with the lower administrative distance. However, an OSPFv2 or OSPFv3 intra-area route is always preferred over an OSPFv2 or OSPFv3 inter-area route, even if the intra-area route's distance is greater than the inter-area route's distance.

The **no** form of the commands reverts to the default setting.

## Examples

The following example sets the distance value for all external routes to 125.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# distance external 125
```

The following example sets the distance value for intra-area routes to 80.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# distance intra-area 80
```

The following example sets the distance value for inter-area routes to 90.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# distance inter-area 90
```

## History

Release version	Command history
5.0.0	Support was added for OSPFv3.

# distribute-list route-map

Creates a route-map distribution list.

## Syntax

```
distribute-list route-map map in
no distribute-list route-map
```

## Parameters

*map*  
Specifies a route map.

**in**  
Creates a distribution list for an inbound route map.

## Modes

OSPF router configuration mode  
 OSPFv3 router configuration mode  
 OSPF router VRF configuration mode  
 OSPFv3 router VRF configuration mode

## Usage Guidelines

The distribution list can filter Link State Advertisements (LSAs) received from other OSPF devices before adding the corresponding routes to the routing table.

The **no** form of the command removes the distribution list.

## Examples

The following example creates a distribution list using a route map named filter1 that has already been configured.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# distribute-list route-map filter1 in
```

## History

Release version	Command history
5.0.0	Support was added for OSPFv3.



# distribute-list prefix-list (OSPFv3)

Applies a prefix list to OSPF for IPv6 routing updates. Only routes permitted by the prefix-list can go into the routing table.

## Syntax

```
distribute-list prefix-list list-name in
no distribute-list prefix-list
```

## Command Default

Prefix lists are not applied to OSPFv3 for IPv6 routing updates.

## Parameters

*list-name*

Name of a prefix-list. The list defines which OSPFv3 networks are to be accepted in incoming routing updates.

**in**

Applies the prefix list to incoming routing updates on the specified interface.

## Modes

OSPFv3 router configuration mode

OSPFv3 VRF router configuration mode

## Usage Guidelines

The **no** form of the command removes the prefix list.

## Examples

The following example configures a distribution list that applies the filterOspfRoutes prefix list globally:

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# distribute-list prefix-list filterOspfRoutes in
```

## History

Release version	Command history
5.0.0	This command was introduced.

# dot1x authentication

Enables 802.1x authentication on a port.

## Syntax

**dot1x authentication**

**no dot1x authentication**

## Command Default

802.1x authentication is disabled for ports.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Port control must be configured to activate authentication on an 802.1x-enabled interface using the **dot1x port-control auto** command from interface configuration mode.

Enter the **no dot1x authentication** command to disable dot1x on the port and remove the configuration from 802.1x management.

## Examples

The following example enables 802.1x authentication on a specific 10-gigabit Ethernet interface port:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# dot1x authentication
```

The following example disables 802.1x authentication on a specific 40-gigabit Ethernet interface port and remove the configuration from 802.1x management:

```
device# configure terminal
device(config)# interface fortygigabitethernet 180/0/6
device(conf-if-fo-180/0/6)# no dot1x authentication
```

# dot1x enable

Enables 802.1X authentication globally.

## Syntax

`dot1x enable`

`no dot1x enable`

## Command Default

Authentication is disabled globally.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no dot1x enable** to disable 802.1X authentication globally.

## Examples

The following example enables 802.1X authentication globally:

```
device(config)# dot1x enable
```

# dot1x mac-auth-bypass

Configures MAC authentication bypass (MAB) to authenticate the client based on the MAC address if the 802.1X authentication times out while waiting for an Extensible Authentication Protocol over LAN (EAPoL) message exchange.

## Syntax

```
dot1x mac-auth-bypass
```

```
no dot1x mac-auth-bypass
```

## Command Default

This feature is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **dot1x reauthMax** command and **dot1x reauthentication** command must be configured to initiate MAB. The maximum number of reauthentication attempts configured using the **dot1x reauthMax** command, must be set to a value from 3 through 10 to initiate MAB. If EAPoL response is received within the specified number of attempts, 802.1X authentication remains active and authentication request is sent to the RADIUS server.

MAB is supported only on Layer 2 switch ports configured as access ports. MAB is not supported on port-channel members, port-channels, profiled port, ISL port, and trunk port.

If the port authentication mode is set to force-authorized or force-unauthorized, the port does not fall back to MAB. That is, the port remains in port-based authentication mode itself.

Use the **no** form of this command to disable MAB.

## Examples

The following example configures MAB on a switch port.

```
device# configure terminal
device(config)# dot1x enable
device(config)# interface tengigabitethernet 5/1/12
device(conf-if-te-5/1/12)# switchport
device(conf-if-te-5/1/12)# dot1x authentication
device(conf-if-te-5/1/12)# dot1x port-control auto
device(conf-if-te-5/1/12)# dot1x reauthentication
device(conf-if-te-5/1/12)# dot1x reauthMax 3
device(conf-if-te-5/1/12)# dot1x mac-auth-bypass
```

## History

Release version	Command history
7.1.0	This command was introduced.

## Related Commands

[dot1x reauthentication](#), [dot1x reauthMax](#)

# dot1x mac-auth-enable

Configures MAC authentication to authenticate the client based on the MAC address.

## Syntax

**dot1x mac-auth-enable**

**no dot1x mac-auth-enable**

## Command Default

This feature is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If 802.1X or MAB is enabled for a port, MAC authentication cannot be enabled on the same port.

MAC authentication is supported only on Layer 2 switch ports configured as access ports. These authentication methods are not supported on port-channel members, port-channels, profiled port, ISL port, and trunk port.

Use the **no** form of this command to disable MAC authentication.

## Examples

The following example configures MAC authentication.

```
device# configure terminal
device(config)# dot1x enable
device(config)# interface tengigabitethernet 5/1/12
device(conf-if-te-5/1/12)# switchport
device(conf-if-te-5/1/12)# dot1x mac-auth-enable
```

## History

Release version	Command history
7.1.0	This command was introduced.

# dot1x port-control

Controls port-state authorization.

## Syntax

```
dot1x port-control { auto | force-authorized | force-unauthorized }  
no dot1x port-control
```

## Command Default

The default port state is **auto**.

## Parameters

### auto

Enables authentication on a port. The controlled port is unauthorized until authentication takes place between the client and authentication server. Once the client passes authentication, the port becomes authorized. This has the effect of activating authentication on an 802.1x-enabled interface.

### force-authorized

Forces a port to remain in an authorized state. This also allows connection from multiple clients.

### force-unauthorized

Forces a port to remain in an unauthorized state.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no dot1x port-control** to return to the default setting.

## Examples

The following example enables the port state to auto on a specific 10-gigabit Ethernet interface port.

```
device# configure terminal  
device(config)# interface tengigabitethernet 178/0/9  
device(conf-if-te-178/0/9)# dot1x port-control auto
```

The following example enables the port state to force-authorized on a specific 40-gigabit Ethernet interface port.

```
device# configure terminal  
device(config)# interface fortygigabitethernet 180/0/1  
device(conf-if-fo-180/0/1)# dot1x port-control force-authorized
```

# dot1x quiet-period

Configures the time interval that the device remains idle between a failed authentication and a reauthentication attempt.

## Syntax

```
dot1x quiet-period seconds  
no dot1x quiet-period
```

## Command Default

The default quiet period is 60 seconds.

## Parameters

*seconds*

Specifies the time between failed reauthentication and reauthentication attempt. Valid values range from 1 through 65535 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Changing the quiet-period interval time to a number lower than the default can result in a faster response time.

The **no dot1x quiet-period** command restores the default setting.

## Examples

The following example changes the interval time to 200 seconds on a specific 10-gigabit Ethernet interface:

```
device(config)# interface tengigabitethernet 178/0/9  
device(conf-if-te-178/9)# dot1x quiet-period 200
```

The following example restores the interval time to the default value on a specific 40-gigabit Ethernet interface:

```
device(config)# interface fortygigabitethernet 180/0/6  
device(conf-if-fo-180/0/6)# no dot1x quiet-period
```



# dot1x reauthenticate

Initiates 802.1X reauthentication on a specified interface.

## Syntax

```
dot1x reauthenticate interface <N> gigabitethernet rbridge-id/slot/port
```

## Parameters

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

The following example initiates reauthentication of a client connected to 10-gigabit Ethernet interface 1/0/16:

```
device# dot1x reauthenticate interface tengigabitethernet 1/0/16
```

# dot1x reauthentication

Configures the device to periodically reauthenticate the clients connected to 802.1X-enabled interfaces at regular intervals.

## Syntax

**dot1x reauthentication**

**no dot1x reauthentication**

## Command Default

Periodic reauthentication is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

When periodic reauthentication is enabled using the **dot1x reauthentication** command, the device reauthenticates the clients every 3,600 seconds by default.

The reauthentication interval is configurable using the **dot1x timeout re-authperiod** command. The reauthentication interval configured using the **dot1x timeout re-authperiod** command takes precedence.

The **no dot1x reauthentication** command disables periodic reauthentication.

## Examples

The following example enables 802.1X reauthentication on a specific 10-gigabit Ethernet interface port:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# dot1x reauthentication
```

The following example disables 802.1X reauthentication on a specific 1-gigabit Ethernet interface port:

```
device# configure terminal
device(config)# interface gigabitethernet 178/2/9
device(conf-if-gi-178/2/9)# no dot1x reauthentication
```

# dot1x reauthMax

Sets the maximum number of times that a port attempts 802.1x reauthentication before the port changes to the unauthorized state.

## Syntax

```
dot1x reauthMax number
```

```
no dot1x reauthMax
```

## Command Default

The number of times that a port attempts 802.1x authentication is 2.

## Parameters

*number*

Specifies the maximum number of reauthentication attempts before the port goes to the unauthorized state. Valid values range from 1 through 10.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no dot1x reauthMax** command restores the default setting.

## Examples

The following example sets the maximum number of reauthentication attempts to 5 on a specific 10-gigabit Ethernet interface port:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# dot1x reauthMax 5
```

The following example sets the maximum number of reauthentication attempts to the default value on a specific 40-gigabit Ethernet interface port:

```
device# configure terminal
device(config)# interface fortygigabitethernet 180/1/9
device(conf-if-fo-180/1/9)# no dot1x reauthMax
```

# dot1x test eapol-capable

Executes the 802.1x readiness check on the switch.

## Syntax

```
dot1x test eapol-capable interface <N> gigabitethernet rbridge-id/slot/port
```

## Parameters

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable.

When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is designated as 802.1x-capable.

The readiness check can be sent on a port that handles multiple hosts (for example, a PC that is connected to an IP phone). The readiness check is not available on a port that is configured with the command **dot1x port-control force-unauthorized**.

The readiness check is typically used before 802.1x is enabled on the switch.

802.1x authentication cannot be initiated while the 802.1x readiness test is in progress.

The 802.1x readiness test cannot be initiated while 802.1x authentication is active.

802.1x readiness can be checked on a per-interface basis. Readiness check for all interfaces at once is not supported.

## Examples

The following example configures readiness check on an interface to determine if the devices connected to the ports are 802.1x-capable.

```
device# dot1x test eapol-capable interface tengigabitethernet 1/0/13
DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on Ten Gigabit Ethernet1/0/13 is EAPOL capable.
```

# dot1x test timeout

Sets the 802.1X readiness test timeout.

## Syntax

`dot1x test timeout timeout`

## Command Default

The default readiness test interval is 10 seconds.

## Parameters

*timeout*

Specifies the readiness test interval value in seconds. Valid values range from 1 through 65535.

## Modes

Global configuration mode

## Examples

The following example sets the test timeout to 30 seconds:

```
device(config)# dot1x test timeout 30
```

## Related Commands

[dot1x test eapol-capable](#)

# dot1x timeout re-authperiod

Sets the number of seconds between reauthorization attempts on a specified interface.

## Syntax

```
dot1x timeout re-authperiod seconds  
no dot1x timeout re-authperiod
```

## Command Default

3600 seconds

## Parameters

*seconds*

Specifies the seconds between reauthorization attempts. Valid values range from 1 through 4294967295 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no dot1x timeout re-authperiod** to return to the default setting.

## Examples

The following example sets 25 seconds as the amount of time between reauthorization attempts on a specific 1-gigabit Ethernet interface:

```
device# configure terminal  
device(config)# interface gigabitethernet 190/0/9  
device(conf-if-gi-190/0/9)# dot1x timeout re-authperiod 25
```

The following example sets the time between reauthorization attempts to the default value on a specific 40-gigabit Ethernet interface:

```
device# configure terminal  
device(config)# interface fortygigabitethernet 180/0/5  
device(conf-if-fo-180/0/5)# no dot1x timeout re-authperiod
```

# dot1x timeout server-timeout

Sets the 802.1X authentication-server response timeout for a specified interface.

## Syntax

```
dot1x timeout server-timeout seconds
```

```
no dot1x timeout server-timeout
```

## Command Default

30 seconds

## Parameters

*seconds*

Specifies the number of seconds that a switch waits for the response from the 802.1X authentication server. Valid values range from 1 through 65535.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no dot1x timeout server-timeout** to return to the default setting.

## Examples

The following example sets 40 seconds as the switch-to-authentication server transmission time on a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# dot1x timeout server-timeout 40
```

The following example sets the switch-to-authentication server transmission time to the default value on a specific 1-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface gigabitethernet 170/4/2
device(conf-if-gi-170/4/2)# no dot1x timeout server-timeout
```



# dot1x timeout supp-timeout

Specifies the EAP response timeout for 802.1X authentication.

## Syntax

```
dot1x timeout supp-timeout seconds  
no dot1x timeout supp-timeout
```

## Command Default

30 seconds

## Parameters

*seconds*

Specifies the number of seconds that the switch waits for a response to the EAP frame. Valid values range from 1 through 65535.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command sets the time in seconds that a switch waits for a response to an Extensible Authentication Protocol (EAP) request frame from the client before resending the request.

Enter **no dot1x timeout supp-timeout** to return to the default setting.

## Examples

The following example sets 45 seconds as the switch-to-client retransmission time for the EAP request frame on a specific 10-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface tengigabitethernet 178/0/8  
device(conf-if-te-178/0/8)# dot1x timeout supp-timeout 45
```

The following example sets the switch-to-client retransmission time for the EAP request frame to the default value on a specific 40-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface fortygigabitethernet 190/0/16  
device(conf-if-fo-190/0/16)# no dot1x timeout supp-timeout
```

# dot1x timeout tx-period

Sets the time the switch waits for a response to an Extensible Authentication Protocol (EAP) request or identity frame.

## Syntax

```
dot1x timeout tx-period seconds  
no dot1x timeout tx-period
```

## Command Default

30 seconds

## Parameters

*seconds*

Specifies the time between successive request ID attempts. Valid values range from 1 through 65535 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command sets the interval between successive attempts to request an ID (EAP ID Req) or identity frame from the client. Enter **no dot1x timeout tx-period** to return to the default settings.

## Examples

The following example sets 34 as the number of seconds to wait for a response to an EAP-request or identity frame from the client before retransmitting the request on a specific 10-gigabit Ethernet interface:

```
device# configure terminal  
device(config)# interface tengigabitethernet 190/0/16  
device(conf-if-te-190/0/16)# dot1x timeout tx-period 34
```

The following example sets the interval between successive attempts to request an ID (EAP ID Req) to the default value on a specific 40-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface fortygigabitethernet 180/0/8  
device(conf-if-fo-180/0/8)# no dot1x timeout tx-period
```

# dpod

Manages Dynamic Ports on Demand (POD) assignments.

## Syntax

```
dpod rbridge-id/slot/port { reserve | release }
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a slot number.

*port*

Specifies a port number.

**reserve**

Reserves a POD assignment for a port that is currently not able to come online but is expected to be viable in the future. A port license assignment that is reserved will be associated with the first port set that has a vacancy.

**release**

Removes a port from the port set to which it is currently assigned.

## Modes

Global configuration mode

## Usage Guidelines

A port POD assignment can only be released if the port is currently offline. Enter **shutdown** to take the port offline.

Do not release a port unless you plan to disconnect the optical link or disable the port persistently. If the link (server or optical) is left in a state where the port could be brought online, the Dynamic POD mechanism will detect this unassigned port and attempt to reassign it to a port set.

This command has no effect on Extreme VDX 8770 devices.

In the Network OS v3.0.0 release this command is supported only on the local device.

## Examples

The following example reserves a POD assignment.

```
device# configure terminal
device(config)# dpod 0/10 reserve
device(config-dpod-0/10)# exit
device(config)# dpod 0/11 reserve
device(config-dpod-0/11)# exit
```

The following example removes a port from a POD port set.

```
device# configure terminal
device(config)# dpod 5/0/10 release
device(config-dpod-5/0/10)# exit
device(config)# dpod 5/0/11 release
device(config-dpod-5/0/11)# exit
```

## Related Commands

[show dpod](#), [show running-config dpod](#)

# dscp-cos

Specifies a user-defined mutation-map to be used on the port.

## Syntax

```
dscp-cos map_name
```

## Parameters

*map\_name*

The user-defined map-name.

## Modes

Policy-map configuration mode

## Usage Guidelines

This command is allowed only for the Ingress direction.

This command can only be configured in for the **class class-default** command.

This command can lead to a possible contradiction if there are other user-defined classes used in the same policy-map which has a set CoS action configured. In this case, defined CoS takes priority over the mutation map.

## Examples

Typical command example:

```
device# configure terminal
device(config)# policy-map mutation
device(config-policymap)# class class-default
device(config-policyclass)# dscp-cos plsmap
```

## Related Commands

[class](#), [policy-map](#)

# dscp-mutation

Specifies the dscp-mutation mutation-map to be used on the port.

## Syntax

```
dscp-mutation map_name
```

## Parameters

*map\_name*

The user-defined map-name.

## Modes

Policy-map configuration mode-

## Usage Guidelines

This command is allowed only for the ingress direction.

This command can only be configured in for the **class class-default** command.

This command can lead to a possible contradiction if there are other user-defined classes used in the same policy-map which has a set cos action configured. In this case-defined cos takes priority over the mutation map.

## Examples

Typical command example:

```
device# configure terminal
device(config)# policy-map mutation
device(config-policymap)# class class-default
device(config-policyclass)# dscp-mutation plsmap
```

## Related Commands

[class](#), [policy-map](#)

# dscp-traffic-class

Specifies the traffic-class mutation-map to be used on the port.

## Syntax

```
dscp-traffic-class map_name
```

## Parameters

*map\_name*

The user-defined map-name.

## Modes

Policy-map configuration mode

## Usage Guidelines

This command is allowed only for the ingress direction.

This command can only be configured in for the **class class-default** command.

This command can lead to a possible contradiction if there are other user-defined classes used in the same policy-map which has a set cos action configured. In this case-defined cos takes priority over the mutation map.

## Examples

Typical command example:

```
device# configure terminal
device(config)# policy-map mutation
device(config-policymap)# class class-default
device(config-policyclass)# dscp-traffic-class plsmap
```

## Related Commands

[class](#), [policy-map](#)

# duplicate-mac-timer

Configures a duplicate MAC detection timer for the detection of continuous MAC moves.

## Syntax

**duplicate-mac-timer** *interval* **max-count** *interval*

**no duplicate-mac-timer** *interval* **max-count** *interval*

## Parameters

*interval*

Specifies the duplicate MAC detection timer interval in seconds. Valid values range from 5 through 300. The default is 5.

**max-count** *value*

Specifies the maximum threshold of MAC moves that can occur within the configured time interval before the MAC address is treated as a duplicate address and further advertisements for that MAC address are blocked. Valid values range from 3 through 10. The default is 3.

## Modes

EVPN instance configuration mode

## Usage Guidelines

The **no** form of the command restores the default values.

## Examples

The following example sets the duplicate MAC detection timer interval to 22 and the maximum count to 5 for EVPN instance "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# duplicate-mac-timer 22 max-count 5
```

The following example restores the default duplicate MAC detection timer and maximum count values.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# no duplicate-mac-timer 22 max-count 5
```

## History

Release version	Command history
7.0.0	This command was introduced.



## Related Commands

[evpn-instance](#)

# ebs

Configures the excess burst size of a class map.

## Syntax

**ebs** *ebs-size*

**no ebs** *ebs-size*

## Parameters

*ebs-size*

Excess burst size. Valid values range from 1250 through 5000000000 bytes in increments of 1 byte.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cls** commands are mandatory for configuring a class map.

If the optional parameters for a class map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

Use the **no** version of this command to remove the parameter from the class map.

## Examples

This example configures a class-map called "default" within a policy map.

```
device# configure terminal
device(config)# policy-map policymap1
device(config-policymap)# class default
device (config-policymap-class)# police cir 40000
device(config-policymap-class-police)# ebs 400000
```

# edge-loop-detection port-priority

Sets the ELD priority for a port.

## Syntax

```
edge-loop-detection port-priority eld-priority
no edge-loop-detection port-priority
```

## Command Default

ELD priority is 128.

## Parameters

*eld-priority*

Specifies the port priority. Valid values range from 0 through 256; a higher number indicates a lower priority.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The ELD priority determines which of the ports involved in a loop will be disabled when the pdu-rx-limit for the data center fabric is reached. The port with the lower priority (higher ELD-priority setting) is the port that is selected to be disabled.

### NOTE

If ELD must select between two ports with the same priority, ELD selects the port with the higher port ID to be disabled.

You must use **edge-loop-detection** to enable edge-loop detection separately on the port for the ELD priority to be effective.

Enter **no edge-loop-detection port-priority** to return to the default setting.

## Examples

The following example sets the ELD priority of a specific 10-gigabit Ethernet interface port.

```
device# configure terminal
device(config)# interface tengigabitethernet 5/0/10
device(cfg-if-te-5/0/10)# edge-loop-detection port-priority 5
```

The following example restores the default ELD priority of 128 to a specific 40-gigabit Ethernet interface port.

```
device# configure terminal
device(config)# interface fortygigabitethernet 8/1/12
device(cfg-if-fo-8/1/12)# no edge-loop-detection port-priority
```

# edge-loop-detection vlan

Enables edge-loop detection (ELD) on a port and VLAN.

## Syntax

```
edge-loop-detection vlan vlan-ID
```

```
no edge-loop-detection vlan vlan-ID
```

## Command Default

Edge-loop detection is disabled.

## Parameters

**vlan** *vlan-ID*

Specifies a VLAN. (Refer to the Usage Guidelines.)

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the VLAN parameter to specify a VLAN and port on which to enable edge-loop detection. The port must be a member of the specified VLAN or the command returns an error.

This functionality detects Layer 2 loops only.

Enter **no edge-loop-detectionvlan** *vlan\_id* to disable edge-loop detection on the specified VLAN.

## Examples

To enable edge-loop detection on VLAN 10 for a specific 10-gigabit Ethernet interface port:

```
device(config)# interface tengigabitethernet 1/0/7
device(conf-if-te-1/0/7)# edge-loop-detection vlan 10
```

To disable edge-loop detection on a specific 1-gigabit Ethernet interface port and a VLAN whose ID is 20:

```
device(config)# interface gigabitethernet 170/1/9
device(conf-if-gi-170/1/9)# no edge-loop-detection vlan 20
```

## eir

Configures the excess information rate for a class map.

## Syntax

**eir** *eir-rate*

**no eir** *eir-rate*

## Parameters

*eir-rate*

Excess information rate. Valid values range from 0 through 40000000000 bps in multiples of 40000.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class map.

If the optional parameters for a class map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

Use the **no** form of this command to remove the parameter from the class map.

## Examples

This example configures a class map called "default" within a policy map.

```
device# configure terminal
device(config)# policy-map policymap1
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# eir 1200000
```

# email

Configures the domain name for Monitoring and Alerting Policy Suite (MAPS) notifications.

## Syntax

```
email {email_address}
no email
```

## Parameters

```
email email_address
```

Destination email address for MAPS notifications. Only five or fewer addresses can be configured.

## Modes

MAPS configuration mode

## Usage Guidelines

Use the **no email** command to remove the email address.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# maps
device(config-rbridge-id-5-maps)# email admin@notwork.biz
device(config-rbridge-id-5-maps)# email alert@notwork.biz
```

## History

Release version	Command history
6.0.1	This command was introduced.

# enable (Access Gateway)

Enables Access Gateway mode on a switch as follows: Enables FC ports, configures them as N\_Ports, and then maps them to VF\_Ports.

## Syntax

`enable`  
`no enable`

## Modes

Access Gateway (AG) configuration mode

## Usage Guidelines

The initial mapping of N\_Ports to VF\_Ports implements the current default mapping configuration.

The **enable** command is not allowed if the RBridge is part of an FCoE forwarder (FCF)-group as an FCF-RBridge ID (rbid) in an FCoE fabric-map configuration.

To disable Access Gateway mode, enter the **no enable** command.

The **no enable** command is not allowed if the RBridge contains a port provisioned for FCoE.

## Examples

The following example enables AG mode on a local switch:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ag
device(config-rbridge-id-1-ag)# enable
```

The following example disables AG mode on a local switch:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ag
device(config-rbridge-id-1-ag)# no enable
```

## History

Release version	Command history
5.0.0	The <b>rbridge-id</b> and <b>vcs-id</b> parameters are no longer supported for this command.
6.0.1	This command name was changed from <b>ag enable</b> to <b>enable</b> , and was moved from privileged EXEC mode to AG configuration mode.

# enable (Fabric-Virtual-Gateway)

Enables IPv4 or IPv6 Fabric-Virtual-Gateway sessions in VCS.

## Syntax

**enable**  
**no enable**

## Command Default

A session under the global VE interface is enabled.

## Modes

Fabric-Virtual-Gateway address-family configuration mode  
Fabric-Virtual-Gateway under VE interface IPv4 or IPv6 configuration mode  
Fabric-Virtual-Gateway on an RBridge VE interface IPv4 or IPv6 configuration mode

## Usage Guidelines

Enter the **no** form of the command to disable a specific IPv4 or IPv6 Fabric-Virtual-Gateway session.  
The session can be enabled at the RBridge-level even if it is disabled at global level.

## Examples

The following example shows how to enable a Fabric-Virtual-Gateway session in Fabric-Virtual-Gateway address-family configuration mode.

```
device(config)# router fabric-virtual-gateway
device(conf-router-fabric-virtual-gateway)# address-family ipv4
device(conf-address-family-ipv4)# enable
```

The following example shows how to enable a Fabric-Virtual-Gateway session in Fabric-Virtual-Gateway in VE interface IPv4 configuration mode.

```
device(config)# interface ve 2000
device(config-ve-2000)# ip fabric-virtual-gateway
device(config-ip-fabric-virtual-gw)# enable
```

The following example shows how to enable a Fabric-Virtual-Gateway session in Fabric-Virtual-Gateway on an RBridge VE interface IPv6 configuration mode.

```
device(config)# rbridge-id 55
device(config-rbridge-id-55)# interface ve 2000
device(config-rbridge-ve-2000)# ipv6 fabric-virtual-gateway
device(config-ipv6-fabric-virtual-gw)# enable
```



## History

Release version	Command history
5.0.1	This command was introduced.

# enable (MAPS)

Enables and sets the policy thresholds for Monitoring and Alerting Policy Suite (MAPS). MAPS policies are designed in a way that thresholds are pre-set to aggressive, moderate, or conservative based on how sensitive the actions are needed.

## Syntax

```
enable { policy polycyname } [ action list value ]
```

```
no enable
```

## Command Default

There is no default policy.

## Parameters

**policy** *polycyname*

- `dft_aggressive_policy` - Contains rules with very strict thresholds, for environments requiring a pristine fabric.
  - `dft_moderate_policy` - Contains rules with thresholds values that lie in-between aggressive and conservative policies.
  - `dft_conservative_policy` - Contains thresholds that are lenient enough to not trigger actions immediately and allows for buffer. This can be used in environments where the elements are resilient and can accommodate for errors.
- Sets the policy level for MAPS. These are the three default policies included in the software.

**action list** *value*

Defines which actions should be taken by the command policy. The action list names are:

- RASLOG
- SNMP
- EMAIL
- SFP\_MARGINAL - Used in context of the Advanced SFP groups only, to set the state of particular SFP as MARGINAL.
- USE-POLICY - Rule actions are retained and used as-is.
- NONE

## Modes

MAPS configuration mode

## Usage Guidelines

Use the **no enable** command to disable MAPS.

When MAPS is disabled then HA fail over will trigger to disable MAPS fully and enable Fabric Watch.

MAPS does not support FlexPort monitoring.

## Examples

Typical command execution. Multiple action values can be used by using commas to separate the values.

```
device(config)# rbridge-id 5
device(config-rbridge-id-5)# maps
device(config-rbridge-id-5-maps)# enable policy dflt_moderate_policy actions email
device(config-rbridge-id-5-maps)# enable policy dflt_conservative_policy actions RASLOG
device(config-rbridge-id-5-maps)# enable policy dflt_moderate_policy actions FENCE,SW_MARGINAL,SNMP
device(config-rbridge-id-5-maps)# enable policy dflt_aggressive_policy actions SW_CRITICAL
```

## History

Release version	Command history
6.0.1	This command was introduced.

# enable (VRRP-E)

Enables a VRRP-E session.

## Syntax

**enable**  
**no enable**

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

Use the **no** form of this command to disable a VRRP-E session.

## Examples

The following example enables a VRRP-E session on VRRP-E group 10 on interface Ve 25.

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# int ve 25
device(config-ve-25)# vrrp-extended-group 10
device(config-vrrp-extended-group-10)# enable
```

## Related Commands

[vrrp-extended-group](#), [vrrp-group](#)

# enable statistics direction

Enables the collection of per-VLAN statistics for VXLAN overlay gateway tunnels.

## Syntax

```
enable statistics direction { both | tx | rx } vlan [ add | remove ] vlan_id
no enable statistics
```

## Parameters

### both

Specifies the collection of statistics for both the receive and transmit directions.

### rx

Specifies the collection of statistics for the receive direction.

### tx

Specifies the collection of statistics for the transmit direction.

### vlan

Specifies a VLAN or range of VLANs to be added or removed for statistics collection.

#### add

Enables statistics collection on a VLAN ID or range of VLAN IDs. You can use this option if you have disabled specific VLAN IDs and now want to re-enable them.

#### remove

Disables statistics collection on a VLAN ID or range of VLAN IDs.

#### *vlan\_id*

A VLAN ID or range of VLAN IDs. The range is from 1 through 4090. See the Usage Guidelines.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

This configuration enables per-VLAN statistics collection for the packets sent and received over the tunnels associated with this gateway instance.

The specified VLAN IDs must already be configured.

If you remove all VLAN IDs from statistics collection, statistics collection becomes disabled and the **remove** option does not appear in the command line interface of the running configuration.

The only way to change the direction once you have executed this command is to enter the **no enable statistics** command, then re-enter the **enable statistics direction** command

The **no** form of this command disables per-VLAN statistics collection for this gateway.

enable statistics direction

You cannot delete an attached VLAN if statistics collection is enabled on that VLAN.

## Examples

The following example enables statistics collection for all VXLAN tunnels in both directions for VLAN IDs 10 and 20 through 30.

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# enable statistics direction both vlan 10, 20-30
```

## Related Commands

[overlay-gateway](#)

# end

Returns to the Privileged EXEC command mode from all configuration modes.

## Syntax

end

## Modes

All configuration modes

## Examples

The following example returns to Privileged EXEC mode from interface configuration mode:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/0
device(conf-if-te-0/0)# end
```

## Related Commands

[exit](#), [interface](#)

# enforce-first-as

Enforces the use of the first autonomous system (AS) path for external BGP (eBGP) routes.

## Syntax

**enforce-first-as**

**no enforce-first-as**

## Command Default

The device does not require the first AS listed in the AS\_SEQUENCE field of an AS path update message from eBGP neighbors be the AS of the neighbor that sent the update.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command disables this feature.

This command causes the router to discard updates received from eBGP peers that do not list their AS number as the first AS path segment in the AS\_PATH attribute of the incoming route.

The device accepts the update only if the AS numbers match. If the AS numbers do not match, the device sends a notification message to the neighbor and closes the session. This requirement applies to all updates received from eBGP neighbors.

## Examples

The following example configures the device to enforce the use of the first AS path.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# enforce-first-as
```



# error-disable-timeout enable

Enables the timer to bring the interface out of the error-disabled state.

## Syntax

```
error-disable-timeout enable
```

## Modes

Spanning tree configuration mode

## Usage Guidelines

When the Spanning Tree Protocol (STP) Bridge Protocol Data Unit (BPDU) guard disables a port, the port remains in the disabled state unless the port is enabled manually. This command allows you to enable the interface from the disabled state.

The command is the same regardless of which type of STP is enabled.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes.

## Examples

To bring the interface out of the disabled state:

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# error-disable-timeout enable

device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# error-disable-timeout enable

device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# error-disable-timeout enable

device# configure terminal
device(config)# protocol spanning-tree pvst
device(conf-pvst)# error-disable-timeout enable

device# configure terminal
device(config)# protocol spanning-tree rpvst
device(conf-rpvst)# error-disable-timeout enable
```

## Related Commands

[error-disable-timeout interval](#)

# error-disable-timeout interval

Sets the timeout interval for errors on an interface.

## Syntax

**error-disable-timeout interval** *seconds*

**no error-disable-timeout interval**

## Command Default

300 seconds

The timeout feature is disabled.

## Parameters

*seconds*

Specifies the time for the interface to time out. Valid values range from 10 through 1000000 seconds.

## Modes

Spanning tree configuration mode

## Usage Guidelines

Enter **no error-disable-timeout interval** to return to the default setting.

The command is the same regardless of which type of STP is enabled.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes.

## Examples

Follow these examples to set the timeout interval.

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# error-disable-timeout interval 100
```

```
device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# error-disable-timeout interval 100
```

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# error-disable-timeout interval 100
```

```
device# configure terminal
device(config)# protocol spanning-tree pvst
device(conf-pvst)# error-disable-timeout interval 100
```

```
device# configure terminal
device(config)# protocol spanning-tree rpvst
device(conf-rpvst)# error-disable-timeout interval 100
```

## esi

Configures an Ethernet Segment (ES) with a unique Ethernet Segment Identifier (ESI) value for a port-channel interface, enabling BGP devices to determine if they are connected to the same ES.

## Syntax

```
esi { value | auto lacp }
```

```
esi { value | auto lacp }
```

## Command Default

None.

## Parameters

*value*

Specifies a 10 byte ESI value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH) associated with an interface.

**auto lacp**

Specifies that the ESI value is automatically derived using LACP.

## Modes

Port-channel configuration mode

## Usage Guidelines

When specifying the 10 byte hexadecimal ESI value, the first and last byte must be a value of 0.

The **no** form of the command deletes the configured ESI for an interface.

## Examples

The following example sets the ESI value for a port-channel interface to 00:22:44:66:88:00:22:44:66:00.

```
device# configure terminal
device(config)# interface port-channel 1
device(config-port-channel-1)# esi 00:22:44:66:88:00:22:44:66:00
```

The following example configures the port-channel configures the port channel to derive the ESI value automatically using LACP.

```
device# configure terminal
device(config)# interface port-channel 1
device(config-port-channel-1)# esi auto lacp
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[interface](#)

# event-handler

Creates or accesses an event-handler profile, which can execute a Python script when a specified trigger occurs.

## Syntax

**event-handler** *event-handler-name* [ **action** **python-script** *file-name* ]

**event-handler** *event-handler-name* [ **description** *description-text* ]

**event-handler** *event-handler-name* [ **trigger** *trigger-id* { **raslog** *raslog-id* [ **pattern** *posix-ext-regex* ] | **vcs** *switch-event* } ]

**no event-handler** *event-handler-name*

## Command Default

No event-handler profile is enabled.

## Parameters

*event-handler-name*

Specifies the name of the event-handler profile. Valid values can have from 1 through 32 characters. The first character must be alphabetic.

**action** **python-script** *file-name*

Specifies a Python file that runs when a trigger-condition occurs. Valid values range from 4 through 32 characters (including the **.py** extension). The first character must be alphanumeric.

**description** *description-text*

Specifies a string describing the event-handler profile. The string can be 1 through 128 ASCII characters in length. Do not use the ? character. If you need to use ! or \, precede each with \.

**trigger** *trigger-id*

Defines an event-handler trigger and specifies an ID number for the trigger. Valid values are 1 through 100, and must be unique per event-handler profile. When the trigger-condition occurs, a Python script is run.

**raslog** *raslog-id*

Specifies a RASlog message ID as the trigger.

**pattern** *posix-ext-regex*

Specifies a POSIX extended regular expression to search for a match within the specified RASlog message ID. For examples, refer to the "trigger" topic.

**vcs** *switch-event*

Specifies a switch event as the trigger. Valid *switch-event* values are as follows:

**switch-bootup**

The switch booted and boot-time configuration is applied.

**switch-ready-for-configuration**

The switch is ready to receive a configuration through an event-handler action.

## Modes

Global configuration mode

Event-handler configuration mode for an existing event handler. (There is no need to enter the **exit** command to return to global configuration mode.)

## Usage Guidelines

You can create multiple event-handler profiles.

You can optionally specify a description, a trigger, or the Python script with this command; or specify them later.

An **event-handler** command creates or accesses an event-handler profile and can also define one of the following parameters:

- Description
- One trigger
- The Python-script action that runs on any trigger

You can also define the above parameters—including one or more triggers—from event-handler configuration mode.

A Python event-handler script runs only if all of the following occur:

- Using the **copy** command, copy the Python file to the `flash://` location on the device.
- Using the **event-handler** command, create an event-handler profile.
- Either using the **event-handler** command or in configuration mode for that profile:
  - Using the **trigger** command, create one or more triggers.
  - Using the **action** command, specify the Python script that will be triggered.
- Using the **event-handler activate** command, activate an instance of the event handler.
- The trigger event occurs.

If an event-handler profile is not activated on any RBridge, the **no** form of this command deletes it.

## Examples

The following example creates an event-handler profile and accesses its configuration mode.

```
device# configure terminal
device(config)# event-handler eventHandler1
device(config-event-handler-eventHandler1)#
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.1.0	The command was modified to support POSIX extended regular expressions for a match within a specified RASlog message ID; and to remove references to fabric cluster mode.

# event-handler abort action

Under Python event-management, aborts a specified event handler that is currently running.

## Syntax

**event-handler abort action** *event-handler-name*

## Parameters

*event-handler-name*

Specifies the name of the event-handler profile. Valid values can have from 1 through 32 characters. The first character must be alphabetic.

## Modes

Privileged EXEC mode

## Examples

The following command successfully aborted event-handler action "eh1".

```
device# event-handler abort action eh1
This operation will abort an event handler action that is currently running and may leave the switch in
an inconsistent state. Do you want to continue? [y/n]:y
Operation completed successfully.
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	The command was changed from <b>clear event-handler action</b> to <b>event-handler abort action</b> .



# event-handler activate

Activates an event handler and accesses event-handler activation mode, from which you can enter advanced configuration commands. You can also append the advanced commands to **event-handler activate**.

## Syntax

**event-handler activate** *event-handler-name*

**event-handler activate** *event-handler-name* [ **action-timeout** *minutes* ] [ **delay** *seconds* ] [ **iterations** *num-iterations* ] [ **interval** *seconds* ] [ **run-mode** *exclusivity-mode* ] [ **trigger-mode** *mode* ] [ **trigger-function** { **OR** | **AND** [ **time-window** *seconds* ] }

**no event-handler activate** *event-handler-name*

## Command Default

No event handler is activated on the device.

## Parameters

*event-handler-name*

Specifies the name of the event-handler profile. Valid values can have from 1 through 32 characters. The first character must be alphabetic.

**action-timeout** *minutes*

Specifies the number of minutes to wait for an action-script to complete execution. If you specify "0", no timeout is set. Valid timeout values are any positive integer.

**delay** *seconds*

Specifies a number of seconds from when a trigger is received until the execution of the specified action begins. Valid values are 0 or a positive integer.

**iterations** *num-iterations*

Specifies the number of times an event-handler action is run, when triggered. Valid values are any positive integer. The default value is 1.

**interval** *seconds*

Specifies the number of seconds between iterations of an event-handler action, if triggered. Valid values are 0 or a positive integer. The default is 0.

**run-mode** *exclusivity-mode*

Specifies if a triggered event-handler action is run in exclusive or non-exclusive mode. The default setting is **non-exclusive**.

**exclusive**

From the triggering of an event-handler action through the completion of the action, cluster formation is not allowed to run. Exclusive run-mode can be applied to configuration-based action scripts where the script can run to completion without cluster formation interrupting configuration NOSCLIs. The exclusive run-mode will hold-off cluster formation operations such as new nodes joining the cluster or existing nodes rejoining the cluster. An active exclusive run-mode will not prevent cluster fail-over operations when the principal node itself is offline and isolated from the cluster and a new principal node is selected.

**non-exclusive**

Cluster formation can occur while a triggered action is in progress.

**trigger-mode** *mode*

Specifies if an event-handler action can be triggered only once or more than once. The default is each time the trigger condition occurs, the event-handler action is launched.

**each-instance**

The event-handler action is launched on each trigger instance received.

**on-first-instance**

As long as the device is running, the event-handler action is launched only once. Following a device restart, the event-handler action can be triggered again.

**only-once**

For the duration of a device's configuration, the event-handler action is launched only once.

**trigger-function**

For an implementation of an event-handler profile, if multiple triggers are defined for an event-handler action, specifies if the action runs only if all of the triggers occur; or if one is sufficient.

**OR**

The event-handler action runs if any of the triggers occur.

**AND**

The event-handler action runs only if all of the triggers occur.

**time-window** *seconds*

In seconds, specify the time window within which all of the triggers must occur in order that the event-handler action runs. Once all triggers have been received and on each subsequent trigger received, the action will be launched when the time difference between the latest trigger and the oldest trigger is less than or equal to the configured time-window.

## Modes

RBridge-ID configuration mode

Global configuration mode

Event-handler activation mode for an existing event handler. (There is no need to enter the **exit** command.)

## Usage Guidelines

You can activate up to 10 different event-handler profiles on a device.

A Python event-handler script runs only if all of the following occur:

- Using the **copy** command, copy the Python file to the `flash://` location on the device.
- Using the **event-handler** command, create an event-handler profile.
- In configuration mode for that profile:
  - Using the **trigger** command, create one or more triggers.
  - Using the **action** command, specify the Python script that will be triggered.
- Using the **event-handler activate** command, activate an instance of the event handler.
- The trigger event occurs.

Following an initial triggering of an event-handler action, any subsequent trigger launches the action an additional time if the following conditions are true:

- The **trigger-mode** parameter is set to the default **each-instance**.
- The subsequent trigger occurs within the specified **time-window**.

For additional usage guidelines regarding the advanced configuration commands, see the following topics:

- **action-timeout**
- **delay**
- **iterations**
- **interval**
- **run-mode**
- **trigger-mode**
- **trigger-function**

To inactivate an event-handler instance on a device, use the **no** form of this command. If an event-handler Python script is running, it is executed to completion before inactivation of the event handler.

## Examples

This example activates eventHandler1 on RBridge 1.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)#
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	This command was modified to support the <b>action-timeout</b> parameter.

# evpn-instance

Configures an Ethernet Virtual Private Network (EVPN) instance.

## Syntax

**evpn-instance** *name*

**no evpn-instance** *name*

## Command Default

An EVPN instance is not configured.

## Parameters

*name*

Specifies an EVPN instance name of up to 32 characters.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Only one EVPN instance is currently supported.

The **no** form of the command removes the configured EVPN instance from the device and removes all configurations under the EVPN instance.

## Examples

The following example creates an EVPN instance called "myinstance" and enters EVPN instance configuration mode.

```
device# configure terminal
device(config)# rbridge-id 11
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)#
```

The following example removes the EVPN instance called "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# no evpn-instance myinstance
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

`duplicate-mac-timer`, `rd auto` (EVPN), `route-target` (EVPN), `vni` (EVPN)

# exceed-set-dscp

Configures the CIR packet IP precedence of a class map.

## Syntax

```
exceed-set-dscp dscp-num
```

```
no exceed-set-dscp dscp-num
```

## Parameters

*dscp-num*

Specifies that traffic with bandwidth requirements that exceed the rate configured for CIR and sent to the EIR bucket will have packet IP precedence set to the value in the *dscp-num* variable. Valid values are 0 through 7.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class map.

If the optional parameters for a class map are not set, they are treated as disabled. To delete parameters for a class-map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

Use the **no** version of this command to remove the parameter from the class map.

## Examples

Example of setting this parameter.

```
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# exceed-set-dscpc 4
```

# exceed-set-prec

Configures the CIR packet IP precedence of a class-map.

## Syntax

```
exceed-set-prec prec-num
```

```
no exceed-set-prec prec-num
```

## Parameters

*prec-num*

Specifies that traffic with bandwidth requirements that exceed the rate configured for CIR and sent to the EIR bucket will have packet IP precedence set to the value in the *prec-num* variable. Valid values are 0 through 7.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class map.

If the optional parameters for a class map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

Use the **no** version of this command to remove the parameter from the class-map.

## Examples

Example of setting this parameter.

```
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# exceed-set-prec 4
```

# exceed-set-tc

Configures the queue assignment of the *trafficclass* variable for a class map.

## Syntax

```
exceed-set-tc trafficclass
```

```
no exceed-set-tc trafficclass
```

## Parameters

*trafficclass*

Specifies that traffic with bandwidth requirements that exceed the rate configured for CIR and is in the limit of what is configured for EIR will have its traffic class (internal queue assignment) set to the value in the *trafficclass* variable. Valid values are 0 through 7.

## Modes

Policy map class police configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class-map.

If the optional parameters for a class map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

Use the **no** version of this command to remove the parameter from the class map.

## Examples

Example of setting this parameter.

```
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class-police)# exceed-set-tc 4
```



# execute-script

Executes a script on the device.

## Syntax

```
execute-script script-name [verbose] [prefix-check] [prefix] [vrf ]
```

## Parameters

*script-name*

The string name of the script to be executed. The valid script names are listed in the Usage Guidelines below.

**verbose**

Forces the script to display the commands being executed as the script is executed.

*prefix-check*

String to verify the structure of the prefix, such as "dddd.eeee.ffff".

*prefix*

The IPv4 address of the target device.

*vrf*

The VRF of the target device.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command applies only to scripts installed along with the firmware. For other scripts, refer to the following resources:

- The **python** topic in the current document
- The "Python Event-Management and Scripting" topic in the *Extreme Network OS Management Configuration Guide*.

The available scripts are:

- `clear_system_counters.py`—Clears all dataplane counters across the system
- `crontab-input`—Monitors for BE link flap
- `db_script`—Wrapper for peek and poke commands
- `ipfabric-egressNode-debugv2.py`—Egress node debug automation script for IP Fabric. This script supports a traditional leaf-spine IP-Fabric topology only.
- `ipfabric-ingressNode-debug.py`—Ingress node debug automation script for IP Fabric. This script supports a traditional leaf-spine IP-Fabric topology only.
- `ipfabric-spineNode-debugv2.py`—Spine node debug automation script for IP Fabric. This script supports a traditional leaf-spine IP-Fabric topology only.
- `rte_cap_acl`—Data-plane debug script
- `wlv_db`—WLV database debug script

The **execute-script** command is supported on the Extreme VDX 6740, 6940, and 8770 platforms.

**NOTE**

Extreme recommends contacting your technical support representative before executing these scripts.

## Examples

The following example displays a typical data trace output for the ipfabric-ingressNode-debugv2.py script. For details, refer to the *Extreme Network OS Troubleshooting Guide* "Tracing the data path with a script" topic.

### NOTE

The ipfabric-ingressNode-debugv2.py script functions under Network OS version 7.0.1a or later only.

```
device# execute-script ipfabric-ingressNode-debugv2.py 50eb.1a81.0857 0067.6866.0010 72.1.77.2/24
62.1.66.2/24 6201

INPUT PARAMETERS:
=====

DA = 50eb.1a81.0857 , SA = 0067.6866.0010 , DIP = 72.1.77.2 , SIP = 62.1.66.2 , Ingress Vlan = 6201

DERIVING & VERIFYING FROM INPUT PARAMETERS:
=====

VDX Platform Model: 6740
Management Interface IP Addresses: ['10.20.234.67/21']

Verified the Vlan is valid & active..further checks..
VE MAC Address = 50eb.1a81.0857
VE Interface config: Anycast IP configured: 62.1.66.1/24 Anycast MAC configured: 0000.6768.0001
Extracting vrf, vni, VE information.....<please wait>..
Appended list of VRF Import RT configured for this VRF: 6610:6210,7710:7200

DERIVED PARAMETERS:
=====

Interface VE IP Address = 62.1.66.1, VRF = vrf10, Int VE MAC addr (My DA list) = ['50eb.1a81.0857',
'0000.6768.0001']
L3VNI = 7200 , L3VNI VE = 7200 VRF Import L3-RT configured for this VRF:
6610:6210,7710:7200
Comment: Flow classified as L3FWD as DA:50eb.1a81.0857 is in MYDA List:['50eb.1a81.0857',
'0000.6768.0001']

VERIFYING MY-DA PROGRAMMED IN HW:
=====

VE MAC programmed in HW 50eb1a810856 matches that in Software 50eb1a810856 ..proceeding
DEBUG: Constructed Anycast MAC programmed for VDX6740 in HW = 000067680001
VE Anycast MAC 000067680001 programmed in HW matches that in Software 000067680001 ..proceeding

CHECK FOR DIP IN LOCAL NODE TO CLASSIFY FLOW:
=====

Route found for DIP: 72.1.77.2
SYMMETRIC ROUTING: It is an EVPN Route with Egress Port VE 7200 being the same as VRF L3VNI VE 7200
ECMP Route 1 : 72.1.77.0/24 EVPN Ve 7200 Bi
DEBUG: 2 Byte AS = 122

CHECKING VNI SYNC BETWEEN HW & SW
=====

Verified the VNI programmed in Software for VLAN 7200 is same as programmed in Hardware

IDENTIFYING THE EVPN BGP PEERS:
=====

Checking BGP table for Peering:
Total Number of BGP Peers (including non-eVPN peers): 3

Details about eVPN BGP peers:
eVPN Local BGP Peering IP: 67.67.67.67 / eVPN Remote BGP Peering IP: 122.122.122.122
```

eVPN Local BGP Peering IP: 67.67.67.67 / eVPN Remote BGP Peering IP: 125.125.125.125

SYMMETRIC FLOW VERIFICATION:

=====

Output derived from show ip route detail:

Path 1 : GW MAC Address: 0005.3365.377b Tunnel Interface: Tu 61441  
 Path 2 : GW MAC Address: 0005.3365.3633 Tunnel Interface: Tu 61441

SYMMETRIC RTG: CHECKING THE BGP VRF TABLES:

=====

Checking BGP VRF table for the Route existence:

Route FOUND in the BGP VRF Table:  
 BGP Best Route: \*>i 72.1.77.0/24 77.77.77.77  
 Non-best Route: \*i 72.1.77.0/24 77.77.77.77

Checking BGP EVPN table for the Route Properties:

Route FOUND in BGP EVPN table..checking further..

Checking BGP EVPN table for the RT values:

Ingress-Node RT string 6610:6210,7710:7200, MATCHES the route RT list ['7710:7200', '122:7200']  
 ...so should be imported into the VRF on Ingress-Node

Egress VNI carried in the route = 7200

Verified the Route VNI 7200 matches the VNI for the tunnel VLAN 7200

(A) CHECK ROUTE-TABLE IN SOFTWARE:72.1.77.2 (DIP check in VRF Routing Table)

=====

Step1: Checking Software Routing Table .....

Route found for DIP: 72.1.77.2  
 Number of ECMP Paths: 2  
 ECMP Route 1 : 72.1.77.0/24 EVPN Ve 7200 200/0  
 ECMP Route 2 : EVPN Ve 7200 Bi  
 Routes found in following Modules / Slots / Linecards: [0]  
 Next-hop Egress Interfaces for this prefix: ['Ve7200', 'Ve7200']

(B) CHECK ROUTE-TABLE IN SLOT: 72.1.77.2

=====

Step2: Checking if route is programmed in specific Hardware Slots .....

Number of Slots = 0  
 Module 0: RIB Route programmed in Hardware : Yes

(C) CHECK ROUTE-TABLE IN HARDWARE: 72.1.77.2

=====

Step3: Checking the hardware LPM & NHOP for required slots.....

CHECKING HARDWARE FOR MODULE: 0  
 (i) Module 0: LPM Entry FOUND in Hardware for Prefix 72.1.77.0/24  
 Module 0 LPM: Egress Interfaces registered in Hardware: ['vlan0.7200', 'vlan0.7200']  
 Less specific (Subnet) prefix programmed as TRAP in HW ..no further nhop checks

LOG-MSG: Reference to NH Entry NOT found in LPM table in Hardware / ASIC, ... possibly due to conversational behavior

CHECK FOR TUNNEL STATES TOWARDS EGRESS NODE:

=====

Tunnels associated in SOFTWARE for the vlan 7200 on this Rbridge:

Tunnel 1 = Tu 61441  
Tunnel 2 = Tu 61443

Tunnels associated in HARDWARE for the vlan 7200 on this Rbridge:

Found following tunnels associated with this vlan in HARDWARE  
Tunnel 61441 Tunnel 61443  
Number of Tunnels programmed in HW are same as in Software

Total list of Tunnels that are Operationally UP on this Rbridge:

Tunnel Number : 61441 Source IP: 66.66.66.66 Destination IP: 77.77.77.77 Oper State : up  
Tunnel Number : 61442 Source IP: 66.66.66.66 Destination IP: 54.54.54.54 Oper State : up  
Tunnel Number : 61443 Source IP: 66.66.66.66 Destination IP: 71.71.71.71 Oper State : up

\*\*\*LOG-MSG: All Operationally UP tunnels are NOT associated with the vlan 7200

QUICK GLANCE SUMMARY OF INGRESS NODE FLOW STRUCTURE:

=====

FLOW Information:

Flow Forward Type : 13fwd-symmetric  
Flow DIP : 72.1.77.2  
Flow Ingress Vlan : 6201  
Flow Ingress VE VRF : vrf10  
Flow Egress Vlan : 7200  
Flow Egress VE VRF : vrf10  
Flow Egress Vlan VNI : 7200

-----

BGP Derived Information:

Flow VRF Import L3 RT : 6610:6210,7710:7200  
Flow VRF Import L2 RT : None  
Flow IP Route found in BGP VRF Table : 72.1.77.0/24  
Flow IP Route found in BGP EVPN Table : 72.1.77.0/24  
Flow MAC Route found in BGP MAC-VRF Table : N/A  
Flow ARP Route found in BGP MAC-VRF Table : N/A  
Flow Egress LeafNodes VTEP IP : 77.77.77.77  
Flow SpineNodes IP's : ['122.122.122.122', '125.125.125.125']

-----

RIB Derived Information:

Flow IP Route found in RIB : 72.1.77.0/24  
Flow Egress Interface : Tu 61441  
Flow Egress Node Gateway MAC Address : ['0005.3365.377b', '0005.3365.3633']

-----

TUNNEL Information: ( Tu 61441 )

Flow Egress Int Tunnel SrcIP : 66.66.66.66  
Flow Egress Int Tunnel DestIP : 77.77.77.77

-----

HARDWARE Programming:

Flow IP ROUTE L3 LPM Fwding Decision : ['Trap']  
Flow IP ROUTE L3 LPM NH HW Fwding Decision : ['None']  
Flow ARP L3 EXM Fwding Decision : N/A  
Flow ARP L3 EXM NH HW Fwding Decision : N/A  
Flow L2 HW Fwding Entry : []

-----

CHECK ALERTS:

Programming Consistency in SW & HW : True  
Number of Alerts found : 0

=====

Paste the below string on following Spine neighbor nodes for further tracing the packet path:

Reference Format: execute-script ipfabric-spineNode-debugv<x>.py <DIP> <DIP-prefix> <ingressNode-  
bgpPeerIP> <egress-vlan> <fwd-type> <destn-Leaf-Node-IP's> <egressVNI> <MAC-Address> <tunnel-  
sip> <tunnel-dip> <RT>

Spine Node (122.122.122.122): execute-script ipfabric-spineNode-debugv2.py 72.1.77.2 72.1.77.0/24  
67.67.67.67 7200 13fwd-symmetric 77.77.77.77 7200 0005.3365.377b 66.66.66.66 77.77.77.77  
6610:6210,7710:7200

Spine Node (122.122.122.122): execute-script ipfabric-spineNode-debugv2.py 72.1.77.2 72.1.77.0/24  
67.67.67.67 7200 13fwd-symmetric 77.77.77.77 7200 0005.3365.3633 66.66.66.66 77.77.77.77

```
6610:6210,7710:7200
```

```
Spine Node (125.125.125.125): execute-script ipfabric-spineNode-debugv2.py 72.1.77.2 72.1.77.0/24
67.67.67.67 7200 l3fwd-symmetric 77.77.77.77 7200 0005.3365.377b 66.66.66.66 77.77.77.77
6610:6210,7710:7200
```

```
Spine Node (125.125.125.125): execute-script ipfabric-spineNode-debugv2.py 72.1.77.2 72.1.77.0/24
67.67.67.67 7200 l3fwd-symmetric 77.77.77.77 7200 0005.3365.3633 66.66.66.66 77.77.77.77
6610:6210,7710:7200
=====
```

The following example displays a typical output for the clear\_system\_counters.py script.

```
device# execute-script clear_system_counters.py
=====
SCRIPT TRIGGERRED TO CLEAR SWITCH-WIDE STATISTICS
=====
CLI: clear lacp counters
    clear counters storm-control
    clear counters all
    clear counters all rbridge-id all
    clear lldp statistics
    clear counters access-list rbridge-id all in
LOG: Cleared counters for lacp, storm-control, interface, lldp and access-lists
LOG: Clearing ASIC counters for all linecards, please wait....
CLI: foscmd /scripts/wlv_db clear
Clearing switch-wide counters completed successfully
```

## History

Release version	Command history
7.0.1a	This command was enhanced.

# exit

Exits the current mode and returns to the previous mode.

## Syntax

**exit**

## Modes

All command modes

## Usage Guidelines

When used in EXEC and Privileged EXEC modes, the **exit** command terminates the session.

## Examples

The following example exits Interface configuration mode and returns to global configuration mode:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/1)# exit
device(config)# exit
```

## Related Commands

[enable \(VRRP-E\)](#), [interface](#)

## export map evpn

Enables route policy for routes exported from a VRF instance to BGP EVPN.

### Syntax

```
export map { map_name } evpn
no export map { map_name } evpn
```

### Command Default

No route policy is exported.

### Parameters

*map\_name*  
Specifies the name of a route map that is configured by means of the **route-map** command.

### Modes

VRF address-family IPv4 unicast configuration mode  
VRF address-family IPv6 unicast configuration mode

### Usage Guidelines

The export of routes must be configured by means of the **route-target (VRF)** command.

The map is configured per Address Family Identifier (AFI)/Subsequent Address Family Identifier (SAFI) per VRF instance. Once configured, the map is applied to all routes in the VRF for the given AFI/SAFI, and only one map can be applied per VRF at a time. When the second map is applied, the first is overwritten.

Use the **no** form of the command to remove the map.

### Examples

To export a route map in IPv4 VRF address-family unicast configuration mode:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf red
device(config-vrf-red)# address-family ipv4 unicast
device(vrf-red-ipv4-unicast)# export map map1 evpn
```

To remove the map:

```
device(vrf-red-ipv4-unicast)# no export map map1 evpn
```



## History

Release version	Command history
7.1.0	This command was introduced.

## Related Commands

[route-target \(VRF\)](#), [ip extcommunity-list](#)

## extend vlan

Configures switchport VLANs for the tunnels to the containing site in VXLAN overlay gateway configurations.

### Syntax

```
extend vlan { add | remove } vlan_id
no extend vlan
```

### Parameters

#### add

Specifies a VLAN ID or range of VLAN IDs to be added to a tunnel.

#### remove

Specifies a VLAN ID or range of VLAN IDs to be removed from a tunnel.

#### *vlan\_id*

A VLAN ID or range of VLAN IDs. See the Usage Guidelines.

### Modes

VXLAN overlay gateway site configuration mode

### Usage Guidelines

The VXLAN Network Identifier (VNI) classification is derived from the "map vlan" configuration of the parent overlay gateway. This command results in the provisioning or unprovisioning of the VLANs. Use the **no extend vlan *vlan\_id*** command to unprovision a VLAN.

All of the VLAN IDs that are specified must be VLANs that have been mapped by means of the **map vlan *vlan\_id* vni *vni*** command on the parent overlay gateway, unless automatic VNI mapping has been enabled by means of the **map vlan vni auto** command.

Use the **no attach vlan *vlan\_id*** command to remove all switchport configurations from the tunnels to the containing site.

### Examples

Use the **no attach vlan *vlan\_id*** command to remove all switchport configurations from the tunnels to the containing

To configure a switchport VLAN and range of VLANs:

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# site mysite
device(config-overlay-gw-gateway1-site-mysite)# extend vlan add 10,20-30
```

# external-lsdb-limit (OSPFv2)

Configures the maximum size of the external link state database (LSDB).

## Syntax

```
external-lsdb-limit value  
no external-lsdb-limit
```

## Parameters

*value*

Maximum size of the external LSDB. Valid values range from 1 through 14913080. The default is 14913080.

## Modes

OSPF router configuration mode  
OSPF router VRF configuration mode

## Usage Guidelines

If you change the value, make sure to save the running-config file and reload the software. The change does not take effect until you reload or reboot the software.

The **no** form of the command restores the default setting.

## Examples

The following example sets the limit of the LSDB to 20000.

```
device# configure terminal  
device(config)# rbridge-id 122  
device(config-rbridge-id-122)# router ospf  
device(config-router-ospf-vrf-default-vrf)# external-lsdb-limit 20000
```

## Related Commands

[database-overflow-interval \(OSPFv2\)](#)

# external-lsdb-limit (OSPFv3)

Configures the maximum size of the external link state database (LSDB).

## Syntax

```
external-lsdb-limit value
no external-lsdb-limit
```

## Parameters

*value*

Maximum size of the external LSDB. Valid values range from 1 through 250000. The default is 250000.

## Modes

OSPFv3 router configuration mode  
OSPFv3 router VRF configuration mode

## Usage Guidelines

If you change the value, you must save the running-config file and reload the software. The change does not take effect until you reload or reboot the software.

The **no** form of command reverts to the default setting.

## Examples

The following example sets the limit of the external LSDB to 15000.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# external-lsdb-limit 15000
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[database-overflow-interval \(OSPFv3\)](#)

# Commands F through O

---

## fabric dport mode

Configures an ISL interface to support static or dynamic diagnostic-port (D\_Port) testing, or disables D\_Port testing support for the interface.

### Syntax

```
fabric dport mode { dynamic | none | static }
```

### Command Default

Dynamic

### Parameters

#### dynamic

Configures the interface to support dynamic D\_Port testing.

#### none

Disables D\_Port testing support for the interface irrespective of the configuration on the other end of the link.

#### static

Configures the interface to support static D\_Port testing.

### Modes

Interface subtype configuration mode

### Usage Guidelines

There is no **no** form of this command.

### Examples

The following example configures D\_Port testing mode as static and starts basic testing automatically on a TenGigabitEthernet interface.

```
device(config)# interface TenGigabitEthernet 52/0/35
device(conf-if-te-52/0/35)# fabric dport mode static
```

The following example configures D\_Port testing mode as dynamic on a TenGigabitEthernet interface.

```
device(config)# interface TenGigabitEthernet 52/0/35
device(conf-if-te-52/0/35)# fabric dport mode dynamic
```

The following example disables D\_Port testing mode on a TenGigabitEthernet interface, irrespective of the configuration on the other end of the link.

```
device(config)# interface te 52/0/35
device(conf-if-te-52/0/35)# fabric dport mode none
```

## History

Release version	Command history
7.0.0	This command was introduced.

# fabric ecmp load-balance

Configures the list of hashing fields.

## Syntax

```
fabric ecmp load-balance [ dst-mac-vid | src-dst-ip | src-dst-ip-mac-vid | src-dst-ip-mac-vid-port | src-dst-ip-port | src-dst-mac-vid | src-mac-vid ]
```

## Parameters

### dst-mac-vid

Configures the command to use destination MAC address and VID-based load balancing.

### src-dst-ip

Configures the command to use source and destination IP address-based load balancing.

### src-dst-ip-mac-vid

Configures the command to use source and destination IP and MAC address and VID-based load balancing.

### src-dst-ip-mac-vid-port

Configures the command to use source and destination IP, MAC address, VID and TCP/UDP port-based load balancing.

### src-dst-ip-port

Configures the command to use source and destination IP and TCP/UDP port-based load balancing.

### src-dst-mac-vid

Configures the command to use source and destination MAC address and VID-based load balancing.

### src-mac-vid

Configures the command to use source MAC address and VID-based load balancing.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use this command to configure the list of fields (in the incoming packets), used for hashing.

## Examples

The following example set the ECMP load balance to use source and destination IP address-based load balancing.

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# fabric ecmp load-balance src-dst-ip
```

## Related Commands

[fabric ecmp load-balance-hash-swap](#)



# fabric ecmp load-balance-hash-swap

Configures how to swap the input fields for load balancing.

## Syntax

```
fabric ecmp load-balance-hash-swap value
```

## Parameters

*value*

The control value. Valid values range from 0x0 through 0xFFFFFFFF.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use this command to swap the input fields before feeding them to the hash function.

The variable value for this command is interpreted as the bitwise control of the 212-bit key. Each bit controls whether two adjacent bits of the key are to be swapped. This 32-bit control value is written to all four hash-swap control registers. This means that this value is replicated in a 32-bit block to form a 106-bit value. A value of 0x0 does not swap any input fields, while a value of 0xffffffff swaps all 106 input bit-pairs.

## Examples

The following example swaps the ECMP input fields for load balancing and swap all 106 input bit-pairs:

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# fabric ecmp load-balance-hash-swap 0xffffffff
```

## Related Commands

[fabric ecmp load-balance](#)

# fabric isl enable

Enables and disables the administration and operational state of an Inter-Switch Link (ISL).

## Syntax

**fabric isl enable**

**no fabric isl enable**

## Command Default

ISL ports are enabled persistently.

## Modes

Interface subtype configuration mode

## Usage Guidelines

No edge port configuration is allowed on an ISL. If the port is connected to another switch when this command is issued, the fabric may reconfigure.

Enter **no fabric isl enable command** to disable the administration and operational state of an inter-switch link (ISL).

When an RBridge is rejoining the cluster, the interface-level configuration is reset to the default values.

## Examples

To enable the administration and operational state of an ISL on a specific 10-gigabit Ethernet interface:

```
device(config)# interface tengigabitethernet 1/0/18
device(config-if-te-1/0/18)# fabric isl enable
```

To disable the administration and operational state of an ISL on a specific 40-gigabit Ethernet interface:

```
device(config)# interface fortygigabitethernet 1/1/15
device(config-if-fo-1/1/15)# no fabric isl enable
```

## Related Commands

[interface, diag setcycle, show diag burninstatus](#)

# fabric login-policy

Allows a user to configure Duplicate WWN login mode for a particular node or switch in a fabric.

## Syntax

```
fabric login-policy { duplicateWWN } { precedence { new-login | old-login } }
```

## Command Default

The default is **old-login**.

## Parameters

### duplicateWWN

This is the only login policy that is currently supported.

### precedence

Specifies a login policy.

### new-login

Configures the new device to log in and clean up the old login.

### old-login

Configures the old device to retain the login and reject the new login.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This is a per-switch configuration. The command determines whether a new device with a Duplicate WWN is allowed to log in on a switch or not. The configuration policy is not supported for device logins across switches in a fabric.

## Examples

The following configures the nondefault option on an RBridge, allowing a new device to log in and clean up the old login.

```
device(config)# rbridge-id-3
device(config-rbridge-id-3)# fabric login-policy
device(config-login-policy)# duplicateWWN
device(config-duplicateWWN)# precedence new-login
device(config-duplicateWWN)#
```

## History

Release version	Command history
6.0.1	This command was introduced.

# fabric neighbor-discovery disable

Disables neighbor discovery for Extreme devices on a per-interface basis so that the Extreme VDX does not bring up its ports in an uncontrolled fashion until the fabric completely forms. This command is needed when an unconditional EtherChannel is configured between the fabric and an end node, usually ESX or Hypervisors, which does not support LACP. If an Extreme VDX brings up its ports unexpectedly, the data traffic may be compromised.

## Syntax

```
fabric neighbor-discovery disable
```

```
no fabric neighbor-discovery
```

## Command Default

Neighbor discovery is enabled by default.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to re-enable neighbor discovery on this interface.

This command resets an interface so that it renegotiates with the link partner.

## Examples

To disable neighbor discovery on a specified tengigabitethernet interface:

```
device# configure
device(config)# interface tengigabitethernet 1/0/18

device(config-if-te-1/0/18)# fabric neighbor-discovery disable
```

Once the fabric has come online, use the **no fabric neighbor-discovery disable** command to reenable neighbor discovery.

```
device# configure
device(config)# interface tengigabitethernet 1/0/18

device(config-if-te-1/0/18)# no fabric neighbor-discovery disable
```

# fabric port-channel

Configures the list of hashing fields for balancing the data load on port-channels.

## Syntax

```
fabric port-channel port_channel_value load-balance [ dst-mac-vid | src-dst-ip | src-dst-ip-mac-vid | src-dst-ip-mac-vid-port | src-dst-ip-port | src-dst-mac-vid | src-mac-vid ]
```

## Parameters

**port\_channel\_value**

Configures the command to use destination MAC address and VID-based load balancing.

**load-balance**

Configures the command to use destination MAC address and VID-based load balancing.

**dst-mac-vid**

Configures the command to use destination MAC address and VID-based load balancing.

**src-dst-ip**

Configures the command to use source and destination IP address-based load balancing.

**src-dst-ip-mac-vid**

Configures the command to use source and destination IP and MAC address and VID-based load balancing.

**src-dst-ip-mac-vid-port**

Configures the command to use source and destination IP, MAC address, VID and TCP/UDP port-based load balancing.

**src-dst-ip-port**

Configures the command to use source and destination IP and TCP/UDP port-based load balancing.

**src-dst-mac-vid**

Configures the command to use source and destination MAC address and VID-based load balancing.

**src-mac-vid**

Configures the command to use source MAC address and VID-based load balancing.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use this command to configure the list of fields (in the incoming packets), used for balancing the load on port-channels.

## Examples

The following example sets the port-channel load balance to use both source and destination IP address-based load balancing:

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# fabric port-channel 10 load-balance src-dst-ip
```

## History

Release version	Command history
4.0.0	This command was introduced.

# fabric route mcast

Sets the multicast priority for the local RBridge in the fabric.

## Syntax

```
fabric route mcast rbridge-id rbridge-id priority priority
```

## Command Default

Priority is 1.

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**priority**

Sets a priority. The highest priority overrides the lowest RBridge ID and becomes the root.

*priority*

Specifies the priority number of the RBridge.

## Modes

Global configuration mode

## Usage Guidelines

The multicast routing information indicates all ports that are members of the multicast distribution tree: ports that are able to send and receive multicast frames. The root of the tree is auto-selected as the switch with the lowest RBridge ID.

## Examples

The following example changes an RBridge multicast priority.

```
device# configure terminal
device(config)# fabric route mcast rbridge-id 45 priority 5
device(config)# exit
device# show running-config fabric route mcast rbridge-id 45 priority
fabric route mcast rbridge-id 45 priority 5
```

## Related Commands

[show fabric route multicast](#), [show fabric route topology](#), [show running-config fabric route mcast](#)

# fabric trunk enable

Enables and disables trunking on a port.

## Syntax

**fabric trunk enable**

**no fabric trunk enable**

## Command Default

Fabric trunking is enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

When the command is executed to update the trunking configuration, the port to which the configuration applies is disabled and subsequently re-enabled with the new trunking configuration. Traffic through the ports may be temporarily disrupted. Enter **no fabric trunk enable** command to disable trunking on a port.

When an RBridge is rejoining the cluster, the interface-level configuration is reset to the default values.

### NOTE

Trunks are not supported between Extreme 8000 and Extreme VDX 8770 switches, or on the 40-G and 100-G interfaces of a VDX 8770 in non-breakout mode. However, trunking is allowed in breakout mode for a 40-G interface on a VDX 8770.

## Examples

To enable a port for trunking on a specific 10-gigabit Ethernet interface port:

```
device(config)# interface tengigabitethernet 1/0/18
device(config-if-te-1/0/18)# fabric trunk enable
```

To disable a port for trunking on a specific 40-gigabit Ethernet interface port:

```
device(config)# interface fortygigabitethernet 8/10/15
device(config-if-fo-8/10/15)# no fabric trunk enable
```

## Related Commands

[interface](#), [show fabric trunk](#)



# fabric-map

Enables FCoE fabric-map configuration mode and associated options for a default fabric map or a specified fabric map.

## Syntax

```
fabric-map {map_name} [advertisement interval interval | fcmmap fcmmap | keep-alive timeout | priority priority | virtual-fabric vfid | vlan vlan_id]
```

```
no fabric-map
```

```
no advertisement
```

```
no keep-alive
```

```
no priority
```

## Command Default

A default fabric map is created during system boot.

## Modes

FCoE configuration mode

## Usage Guidelines

An FCoE fabric map is equivalent to an FC Virtual Fabric.

The default fabric map configured during system boot appears as follows:

```
!
fcoe
 fabric-map default
   vlan 1002
   priority 3
   virtual-fabric 128
   fcmmap 0E:FC:00
   advertisement interval 8000
   keep-alive timeout
!
```

The user can recreate this if needed by entering the literal string "default" as the value for *map\_name*.

### NOTE

The default VLAN for FCoE is 1002. This value is not set for a nondefault fabric map, and the VLAN ID appears as 0 (invalid). This VLAN ID must be edited to a value other than 1002.

## Examples

To specify the default fabric map, enter the literal string "default" after the **fabric-map** command:

```
device# configure terminal
device(config)# fcoe
device(config-fcoe)# fabric-map default
device(config-fcoe-fabric-map-default)#
```

To specify and configure a nondefault fabric map for a SAN named SanA:

```
device# configure terminal
device(config)# fcoe
device(config-fcoe)# fabric-map SanA
device(config-fcoe-fabric-map-SanA)# advertisement interval 30
device(config-fcoe-fabric-map-SanA)# fcmap 0e:fc:0f
device(config-fcoe-fabric-map-SanA)# no keep-alive
device(config-fcoe-fabric-map-SanA)# no advertisement
device(config-fcoe-fabric-map-SanA)# priority 5
device(config-fcoe-fabric-map-SanA)# virtual-fabric 3000
device(config-fcoe-fabric-map-SanA)# vlan 1010
```

To delete the SAN named SanA:

```
device# configure terminal
device(config)# fcoe
device(config-fcoe)# no fabric-map SanA
```

## History

Release version	Command history
6.0.0	This command was modified to support nondefault fabric maps.

## Related Commands

[fcoe](#)

# fast-external-fallover

Resets the session if a link to an eBGP peer goes down.

## Syntax

```
fast-external-fallover  
no fast-external-fallover
```

## Modes

BGP configuration mode

## Usage Guidelines

Use this command to terminate and reset external BGP sessions of a directly adjacent peer if the link to the peer goes down, without waiting for the timer, set by the BGP **timers** command, to expire. This can improve BGP convergence time, but can also lead to instability in the BGP routing table as a result of a flapping interface.

## Examples

The following example configures the device to reset the session if a link to an eBGP peer goes down.

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# router bgp  
device(config-bgp-router)# fast-external-fallover
```

## Related Commands

[timers \(BGP\)](#)

# fastboot

Reboots the control processor (CP), bypassing the power-on self-tests (POST).

## Syntax

**fastboot**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command performs a "cold reboot" (power off and restart) of the control processor, bypassing POST when the system comes back up. Bypassing POST can reduce boot time significantly.

The **fastboot** operation is disruptive, and the command prompts for confirmation before executing. When you reboot a switch connected to a fabric, all traffic to and from that switch stops. All ports on that switch remain inactive until the switch comes back online.

On a modular chassis, the **fastboot** commands only reboots the management module on which the command is executed. If you log in to the switch IP address and execute the fastboot command, only the active management module reboots and POST is bypassed.

## Examples

The following example performs a cold reboot on the device.

```
device# fastboot
Are you sure you want to fastboot the switch [y/n]?: y
```

## Related Commands

[reload](#)

# fcf-group

Configures a FCoE Forwarder (FCF) group on the Access Gateway (AG) and enters FCoE FCF group configuration mode. The FCF supports FCoE edge ports or virtual Ethernet (VE) ports.

## Syntax

```
fcf-group group_name { fcf-rbid rbid | fif-rbid { add rbridge-id | remove rbridge-id } }
no fcf-group
```

## Parameters

*group\_name*

User-specified (nondefault) name of a configured FCF group.

**fcf-rbid***rbid*

Specifies the RBridge ID of the AG functioning as the FCF.

**fif-rbid**

Adds or removes an RBridge ID from the map.

**add**

Adds one or more RBridge IDs. Comma delimiters and ranging (with a hyphen) are allowed. The RBridge ID range is from 1 through 239. Up to a maximum of 537 characters is allowed in the configuration line.

**remove**

Removes one or more RBridge IDs. Comma delimiters and ranging (with a hyphen) are allowed.

## Modes

FCoE configuration mode

FCoE fabric-map configuration mode

FCoE FCF group configuration mode

## Usage Guidelines

An FCF group is allowed on any switch; an Access Gateway is not required. However, to configure the **fcf-rbid** command on a node, that node must be in Access Gateway mode.

Use the **no** form of this command in FCoE configuration mode to remove the FCF group.

## Examples

The following enters FCoE configuration mode, followed by fabric-map configuration mode, and finally FCoE FCF group configuration mode.

```
device(config)# fcoe
device(config-fcoe)# fabric-map SanA
device(config-fcoe-fabric-map-SanA)# fcf-group SanA
device(config-fcoe-fabric-map-SanA-fcf-group-SanA)# fcf-rbid 10
device(config-fcoe-fabric-map-SanA-fcf-group-SanA)# fif-rbid add 20,30,40
device(config-fcoe-fabric-map-SanA-fcf-group-SanA)# fif-rbid remove 20,30,40
```

To delete an FCF group called "rack-1":

```
device(config)# fcoe
device(config-fcoe)# fabric-map SanA
device(config-fcoe-fabric-map-SanA)# no fcf-group rack-1
```

## History

Release version	Command history
6.0.0	This command was introduced.
6.0.1	This command was modified.

# fcmap

Configures the Fabric Provided MAC Address (FPMA) FCoE MAC Address Prefix (FCMAP) value for an FCoE fabric-map.

## Syntax

```
fcmap hh:hh:hh
```

## Command Default

The default FPMA FCMAP value is 0E:FC:00.

## Parameters

*hh:hh:hh*

A valid FPMA FCMAP value. Valid values range from 0E:FC:00 through 0E:FC:FF.

## Modes

FCoE fabric-map configuration mode

## Usage Guidelines

You must be in FCoE fabric-map configuration mode for this command to function.

## Examples

To configure a nondefault FPMA FCMAP value for the fabric map "SanA":

```
device# configure terminal
device(config)# fcoe
device(config-fcoe)# fabric-map SanA
device(config-fcoe-fabric-map-SanA)# fcmap 0E:FC:01
```

## Related Commands

[fcoe](#), [fabric-map](#)

# fcoe

Enables the FCoE configuration mode.

## Syntax

`fcoe`

## Modes

Global configuration mode

## Examples

```
device# configure terminal
device(config)# fcoe
device(config-fcoe)#
```

## Related Commands

[fabric-map](#)



# fcoe binding

Converts dynamic (automatic) bindings of logical FCoE ports to Ethernet ports to static bindings.

## Syntax

```
fcoe binding { convert-to-static mac-address [ all ]
```

## Command Default

When an FCoE login is serviced on an Ethernet port, a logical FCoE port is bound automatically by default to the Ethernet port, in what is referred to as "dynamic binding."

## Parameters

### convert-to-static mac-address

Converts FCoE interface MAC-address port bindings from dynamic bindings to static bindings.

### all

Converts all FCoE interface MAC-address port bindings from dynamic bindings to static bindings.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command does not have a **no** form.

## Examples

To convert all FCoE interface MAC-address port bindings from dynamic bindings to static bindings.

```
device# fcoe binding convert-to-static mac-address all
```

## History

Release version	Command history
6.0.0	This command was introduced.

# fcoe-enodes

Sets the number of FCoE ENodes that are to be created on a switch.

## Syntax

`fcoe-enodes number`

## Command Default

The default value is 64.

## Parameters

*number*

The number of FCoE interfaces. The range is from 0 through 1000.

## Modes

RBridge ID configuration mode

FCoE configuration mode

## Usage Guidelines

This feature requires an FCoE license. If that license is not present, the number of FCoE ENodes created is 0. When that license is removed, it is recommended that the switch be rebooted. The number of FCoE ENodes created is set to 0 and all interfaces are deleted.

## Examples

The following example sets the number of FCoE ENodes to be created to 67.

```
device# configure terminal
device(config)# rbridge-id 10
sw0(config-rbridge-id-10)# fcoe
sw0(config-rbridge-fcoe)# fcoe-enodes 67
2013/08/16-09:59:11, [FCOE-1035], 9267, DCE, INFO, sw0, Virtual FCoE port 1/19/65 is online.
2013/08/16-09:59:11, [FCOE-1035], 9268, DCE, INFO, sw0, Virtual FCoE port 1/19/66 is online.
2013/08/16-09:59:11, [FCOE-1035], 9269, DCE, INFO, sw0, Virtual FCoE port 1/19/67 is online.
device(config-rbridge-fcoe)#
```

# fcoe-profile (AMPP)

Activates the FCoE profile configuration mode for AMPP.

## Syntax

```
fcoe-profile
```

## Modes

Port-profile configuration mode

## Usage Guidelines

Nondefault FCoE profiles are supported for multiple SANs. Refer to "FCoE logical SAN overview" and "Configuring FCoE logical SANs" in the *Extreme Network OS Layer 2 Switching Configuration Guide*.

The FCoE profile configuration mode for AMPP allows configuration of the FCoE attributes of a port-profile.

## Examples

```
device# configure terminal
device(config)# port-profile default
device(config-port-profile-default)# fcoe-profile
device(config-fcoe-profile)# fcoeport default
```

# fcoeport

Configures a specific port to be an FCoE port with the fabric-map specified by the assigned fabric-map name.

## Syntax

```
fcoeport map-name[ns-ip-registration]
no fcoeport
```

## Parameters

*map-name*

Name of the FCoE fabric map.

**ns-ip-registration**

Configures the interface to either accept or reject the RIP\_NN request. By default, the request is rejected.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to configure a specific port to be an FCoE port with the fabric map specified by the assigned fabric-map name.

Enter "default" for the *map-name* to specify a default fabric map.

Enter **no fcoeport** to remove the FCoE fabric map.

## Examples

To provision a specific 10-gigabit Ethernet interface port with the FCoE fabric map "SanA":

```
device(config)# interface tengigabitethernet 101/0/1
device(conf-if-te-101/0/1)# fcoeport SanA
```

## History

Release version	Command history
6.0.0	This command was modified to add support for any user-specified FCoE fabric map.
6.0.2	This command was modified to support the <b>ns-ip-registration</b> keyword.

## Related Commands

[interface](#)

# fcport-group

Enables the mapping of multiple FCoE ports to a specified VLAN in support of FCoE logical SANs.

## Syntax

**fcport-group**

## Command Default

None

## Modes

FCoE fabric map configuration mode

## Usage Guidelines

Use this command to enable FCoE fabric map fcport-group configuration mode and execute additional commands.

## Examples

This example enables FCoE fabric map fcport-group configuration mode for fabric map "SanA" and lists the available options.

```
device(config)# fcoe
device(config-fcoe)# fabric-map SanA
device(config-fcoe-fabric-map-SanA)# fcport-group
device(config-fcoe-fabric-map-fcport-group-SanA)# ?
Possible completions:
  describe      Display transparent command information
  do             Run an operational-mode command
  exit          Exit from current mode
  fcport-group-rbid  Configure an FCPORT group rbridge-ids.
  help          Provide help information
  no            Negate a command or set its defaults
  pwd           Display current mode path
  top           Exit to top level and optionally run command
```

## History

Release version	Command history
6.0.1	This command was introduced.

# fcport-group-rbid

Enables the user to map multiple local FC ports to a specified VLAN.

## Syntax

```
fcport-group-rbid { [ add | remove ] rbridge-id { rbridge-id }
no fcport-group-rbid rbridge-id { rbridge-id }
```

## Command Default

None

## Parameters

### add

Adds one or more FC port groups to an RBridge ID. Ranging and comma delimiters are allowed.

### remove

Removes one or more FC port groups from an RBridge ID. Ranging and comma delimiters are allowed.

### rbridge-id *rbridge-id*

Specifies an RBridge.

## Modes

FCoE fabric map FC port group configuration mode

## Usage Guidelines

## Examples

This example adds RBridge ID 29 to FC port group "SanA".

```
device(config)# fcoe
device(config-fcoe)# fabric-map SanA
device(config-fcoe-fabric-map-SanA)# fcport-group
device(config-fabric-map-fcport-group-SanA)# fcport-group-rbid add 29
```

## History

Release version	Command history
6.0.1	This command was introduced.

# fcsp auth

Configures the protocol specific parameters.

## Syntax

```
fcsp auth auth-type dh-chap group { 0 | 1 | 2 | 3 | 4 | * } hash { sha1 | md5 | all } policy switch { on | off | active | passive }
```

## Parameters

### auth-type dh-chap

Authentication type is DH-CHAP.

### group

Specifies the DH group value. This parameter sets the strength of the secret. Values are 0, 1, 2, 3, 4 or \*. The asterisk (\*) indicates all values (0 through 4). The default value is \*.

### hash

Specifies the hash type used for authentication. Possible values are **sha1**, **md5**, or **all** (sha1 and md5). The default option is **all**.

### policy switch

Configures the switch authentication policy attribute. Values are **on**, **off**, **passive**, or **active**. The default switch policy is **passive**.

## Modes

Global configuration mode

RBridge ID configuration mode

## Usage Guidelines

This command configures the authentication policy attributes and controls the policy behavior. The policy configuration includes protocol specific parameters such as authentication type, DH-group value, and hash type. It also defines whether the policy is enabled or disabled and how strictly it is enforced.

The authentication policy can be set to any of these values:

- **ON** — Strict authentication is enforced on all E-ports. The ISL goes down (port disable) if the connecting switch does not support the authentication or the policy is OFF. During switch initialization, authentication is initiated on all E-ports automatically. The authentication is initiated automatically during the E-port bring-up by fabric module. The authentication handshaking is completed before the switches exchange the fabric parameters (EFP) for E-port bring-up.
- **ACTIVE** — In this policy, the switch is more tolerant and can be connected to a switch with any type of policy. During switch initialization, authentication is initiated on all E-ports, but the port is not disabled if the connecting switch does not support authentication or the authentication policy is OFF. The authentication is initiated automatically during the E\_Port bring-up.
- **PASSIVE (default)** — The switch does not initiate authentication, but participates in authentication if the connecting switch initiates authentication. The switch does not start authentication on E\_Ports, but accepts the incoming

authentication requests, and will not disable if the connecting switch does not support authentication or the policy is OFF.

- OFF — The switch does not support authentication and rejects any authentication negotiation request from neighbor switch. A switch with the policy OFF should not be connected to a switch with policy ON, since the ON policy is strict and disables the port if any switch rejects the authentication. DH-CHAP shared secrets should be configured before switching on the policy from OFF state.

After the authentication negotiation succeeds, the DH-CHAP authentication is initiated. If DH-CHAP authentication fails, the port is disabled. This behavior applies to all modes of the policy.

Authentication policy configuration is not distributed across the cluster. The rbridge-id of the node should be used to configure protocol and policy configurations.

## Examples

### NOTE

This command is not distributed across the cluster. The RBridge ID of the node should be used to configure protocol and policy configurations.

The following example sets only the authentication type.

```
device# configure terminal
device(config)# rbridge-id 3
device(config-rbridge-id-3)# fcsp auth auth-type dh-chap
```

The following example sets only the hash type.

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# fcsp authhashsha1
```

The following example activates the device authentication policy.

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# fcsp authpolicy switch active
```

## Related Commands

[fcsp auth-secret dhchap](#), [show fcsp auth-secret dh-chap](#), [show running-config fcsp auth](#)



# fcsp auth-secret dhchap

Sets the shared secret for DH CHAP authentication.

## Syntax

```
fcsp auth-secret dhchap node switch_wwn peer-secret secret_key local-secret secret_key
no fcsp auth-secret dhchap node switch_wwn
```

## Parameters

**node** *switch\_wwn*

Specifies the world wide name (WWN) of the peer.

**peer-secret** *secret\_key*

Specifies the secret of the peer that authenticates the peer to the local switch.

**local-secret** *secret\_key*

Specifies the local secret that authenticates the local switch to the peer.

## Modes

Privileged EXEC mode

## Usage Guidelines

The local and peer secret keys must be between 8 and 40 characters long.

Only the following non-alphanumeric characters are valid for the secret key:

```
@ $ % ^ & * ( ) _ + - < > { } [ ] ; ' :
```

This command is supported on the Extreme VDX 6740.

Enter **no fcsp auth-secret dhchap nodeswitch\_wwn** to remove the DHCHAP authentication secrets from the data base, so that the switch with the given WWN cannot connect to the local switch.

## Examples

The following example sets the shared secret for DH-CHAP authentication.

```
device# fcsp auth-secret dhchap node 50:00:51:ed:2d:c0:1e:64 \peer-secret 12345678 local-secret 87654321
```

The following example removes the DH-CHAP authentication secrets from the database.

```
device# no fcsp auth-secret dh-chap node 50:00:51:ed:2d:c0:1e:64
```

```
Shared secret successfully removed
```

## Related Commands

[fcsp auth](#), [show fcsp auth-secret dh-chap](#), [show running-config fcsp auth](#)

# filter-change-update-delay

Changes the delay in the filter-change status prompt from the default.

## Syntax

```
filter-change-update-delay delay_time  
no filter-change-update-delay
```

## Command Default

The default value is 10.

## Parameters

*delay\_time*  
The delay, in seconds, in the filter-change status prompt. Range is from 0 through 600.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Enter 0 (zero) or use the **no** form of this command to disable the timer.

## Examples

The following example changes the delay to 8 seconds.

```
device# configure terminal  
device(config)# rbridge-id 2  
device(config-rbridge-id-2)# filter-change-update-delay 8
```

# fips selftests

Enables Federal Information Processing Standards (FIPS) self tests which will be performed when the switch boots. If the tests run successfully, the switch comes up in the FIPS compliant state.

## Syntax

`fips selftests`

## Command Default

The switch operates in the non-FIPS compliant state.

## Modes

Privileged EXEC mode

## Usage Guidelines

The FIPS self tests include known answer tests (KATs) that exercise various features of FIPS algorithms and conditional tests that test the randomness of random number generators and check for signed firmware. These tests run when the switch boots. Successful completion of these tests places the switch into the FIPS-compliant state. If any test returns an error, the switch reboots and runs the tests again. Whether tests succeed or fail, you cannot return the switch to the non-FIPS compliant state.

Under normal operation, this command is hidden to prevent accidental use. Enter the **unhide fips** command with password "fibranne" to make the command available.

You typically use this command after disabling non-FIPS compliant features on the switch and configuring secure ciphers, but before zeroizing the switch with the **fips zeroize** command. These non-FIPS compliant features that must be disabled include the Boot PROM, root access, TACACS+ authentication, and the dot1x feature. Secure ciphers that must be configured are for the SSH protocol and (optionally) for the Lightweight Directory Access Protocol (LDAP) protocol. The **fips zeroize** command erases all critical security parameters and reboots the switch. Refer to *Network OS Administrator's Guide* for details about preparing a switch for FIPS compliance.

This command can be entered only from a user account with the admin role assigned.



### CAUTION

This command should be used only by qualified personnel. Once a switch is in the FIPS-compliant state, you cannot return it to the non-FIPS compliant state.

## Examples

To enable the FIPS self tests:

```
device# unhide fips
Password: *****
device# fips selftests
Self tests enabled
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

# fips zeroize

Removes all critical security parameters from a switch in readiness for compliance with Federal Information Processing Standards (FIPS) and reboots the switch.

## Syntax

```
fips zeroize
```

## Command Default

The switch operates in the non-FIPS compliant state.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command erases all critical security parameters from the switch in readiness for FIPS compliance including passwords, shared secrets, and private keys. This command also reboots the switch. If FIPS self tests are enabled and they run successfully during reboot, then the switch comes up in the FIPS-compliant mode. If the FIPS self tests return errors, the switch reboots and runs the tests again.

Under normal operation, this command is hidden to prevent accidental use. Enter the **unhide fips** command with password "fibranne" to make the command available.

Typical use of this command is after disabling non-FIPS compliant features, configuring secure ciphers, and enabling FIPS self tests with the **fips selftests** command. These non-FIPS compliant features that must be disabled include the Boot PROM, root access, TACACS+ authentication, and the dot1x feature. Secure ciphers that must be configured are for the SSH protocol and (optionally) for the Lightweight Directory Access Protocol (LDAP) protocol. Refer to the *Extreme Network OS Security Configuration Guide* for details about preparing a switch for FIPS compliance.

This command can be entered only from a user account with the admin role assigned.



### CAUTION

This command should be used only by qualified personnel. Once a switch is in the FIPS-compliant state, you cannot return it to the non-FIPS compliant state.

## Examples

To erase all critical security parameters from a switch:

```
device# unhide fips
```

```
Password: *****
```

```
device(config)# fips zeroize
```

```
This operation erases all passwords, shared secrets, private keys etc. on the switch. Do you want to
continue? [yes,NO] yes
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

# firmware activate

Activates the firmware in the local or remote nodes after installing the firmware that was downloaded with `firmware download noactivate` command.

## Syntax

```
firmware activate [ rbridge-id { rbridge-id } | all ]
```

## Command Default

Activation of the firmware is performed manually by default after a download.

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

### **noactivate**

Downloads the firmware without activating it.

## Modes

Privileged EXEC mode

## Usage Guidelines

By default, the **firmware download** command downloads the firmware to the system, reboots the system, and commits the firmware automatically. You can specify the **noactivate** parameter to download the firmware to the system without activating it (the node is not rebooted). The user can run the **firmware activate** command later to activate the firmware.

## Examples

The following example activates firmware on switch nodes 1, 2, 3, and 5.

```
device# firmware activate rbridge-id rid1-rid3,rid5
```

## Related Commands

[firmware download](#), [firmware restore](#), [show firmwaredownloadstatus](#)



# firmware commit

Commits a firmware upgrade.

## Syntax

```
firmware commit
```

## Modes

Privileged EXEC mode

## Usage Guidelines

The **firmware download** command updates the secondary partitions only. When the **firmware download** command completes successfully and the switch reboots, the system swaps partitions. The primary partition (with the previous firmware) becomes the secondary partition, and the secondary partition (with the new firmware) becomes the primary partition.

By default, **firmware download** automatically commits the firmware after the switch reboots. If you disable auto-commit mode when running **firmware download**, you must execute **firmware commit** to commit the new firmware to the secondary partition.

This command is supported only on the local management modules.

You must run the **firmware download** command with the **nocommit** parameter set for the following firmware commit operation to succeed.

## Examples

The following example commits the firmware.

```
device# firmware commit

Validating primary partition...
Doing firmwarecommit now.
Please wait ...
Replicating kernel image
.....
FirmwareCommit completes successfully.
```

## Related Commands

[firmware downloadshow version](#)

# firmware download

Downloads the firmware on the local switch.

## Syntax

```
firmware download { default-config | ftp | scp | sftp | usb | interactive } [ manual ] [ nocommit ] [ noreboot ] [ noactivate ]
[ coldboot ] host { hostname | host_ip_address } user username password password directory directory [ file file_name ]
[ vcs-mode vcsmode ] [ vcs-id vcsID ] [ rbridge-id rbridge-id ] [ use-vrf vrf-name ]
```

## Command Default

By default, **firmware download** downloads the firmware to the system, reboots the system, and commits the firmware automatically. The user can specify **noactivate** to download the firmware to the system without activating it (the node is not rebooted). You can run the **firmware activate** command later to activate the firmware.

## Parameters

### default-config

Sets the configuration back to default except for the following parameters: VCS mode, VCS ID, and RBridge ID. These three parameters are retained except when the options to change their values are specified.

### ftp | scp | sftp | usb

Valid protocols are **ftp** (File Transfer Protocol) or **scp** (Secure Copy), **sftp** (SSH File Transfer Protocol), **usb** (universal serial bus). The values are not case-sensitive.

### interactive

Runs firmware download in interactive mode. You are prompted for input.

### manual

Updates a single management module in a chassis with two management modules. You must log in to the management module through its dedicated management IP address. This parameter is ignored when issued on a Top-of-Rack (ToR) switch or in a chassis with only one management module.

### nocommit

Disables auto-commit mode. When auto-commit mode is disabled, firmware is downloaded only to the primary partition. You must execute the firmware **commit** command manually to propagate the new image to the secondary partition.

### noreboot

Disables auto-reboot mode. When auto-reboot mode is disabled, you must reboot the switch manually.

### noactivate

Downloads the firmware to the system without activating it, so the node is not automatically rebooted. You can run the **firmware activate** command later to activate the firmware.

### coldboot

Downloads the firmware to the system and reboots both the active and standby MMs.

### host

Specifies the host by DNS name or IP address.

*hostname*

Specifies an IPv4 DNS host name.

*host\_ip\_address*

Specifies the host IP address. IPv4 and IPv6 addresses are supported.

**directory** *directory*

Specifies a fully qualified path to the directory where the firmware is located.

**file** *file\_name*

Specifies the firmware .plist file. This parameter is optional; if unspecified, the default file, release.plist, is used.

**user** *username*

Specifies the user login name for the host.

**password** *password*

Specifies the account password.

**vcs-mode** *vcsmode*

Specifies the new VCS mode. If not set, the existing VCS mode is used. It is only available in local firmware download.

**vcs-id** *vcsID*

Specifies the new VCS ID. If not set, the existing VCS ID is used. It is only available in local firmware download.

**rbridge-id** *rbridge-id*

Specifies the new RBridge ID. If not set, the existing RBridge ID is used. It is only available in local firmware download. ,

**use-vrf** *vrf-id*

Use this option to specify the name of the VRF where the host is located. If this option is not set, mgmt-vrf is used by default.

## Modes

Privileged EXEC mode

## Usage Guidelines

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config.

By default, if you enter the firmware download command without any options, the command invokes ISSU to upgrade the entire system. ISSU involves an High Availability failover of the active management module and is non-disruptive. In contrast, both of the coldboot and default-config options involve system reboots and are disruptive to traffic.

In addition, default-config causes the loss of configuration because it resets the configuration back to the default settings during the firmware upgrade process.

If the **firmware download** command is interrupted because of an unexpected reboot, such as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before beginning another firmware download.

## Examples

Example of firmware download without options (ISSU):

```
device# firmware download ftp directory /buildsjc/sre/SQA/nos/nos6.0.0/nos6.0.0 host 10.31.2.27 user
releaseuser password releaseuser
```

Performing system sanity check...

This command will use the ISSU protocol to upgrade the system. It will cause a WARM reboot and will require that existing telnet, secure telnet or SSH sessions be restarted.

Do you want to continue? [y/n]y

Example of firmware download with the coldboot option:

```
device# firmware download ftp directory /buildsjc/sre/SQA/nos/nos6.0.0/nos6.0.0 host 10.31.2.27 user
releaseuser password releaseuser coldboot
```

Performing system sanity check...

This command will cause a cold/disruptive reboot and will require that existing telnet, secure telnet or SSH sessions be restarted.

Do you want to continue? [y/n]y

Example of firmware download with the default-config option:

```
device# firmware download default-config ftp directory /buildsjc/sre/SQA/nos/nos6.0.0/nos6.0.0 host
10.31.2.27 user releaseuser password releaseuser
```

Performing system sanity check...

This command will cause a cold/disruptive reboot and will require that existing telnet, secure telnet or SSH sessions be restarted.

Do you want to continue? [y/n]y

## History

Release version	Command history
7.0.0	This command entry was enhanced.

## Related Commands

[show version](#)

# firmware download ftp

Specifies FTP as the protocol used to perform a firmware download.

## Syntax

```
firmware download ftp [ coldboot ] [ manual ] [ noactivate ] [ nocommit ] [ noreboot ] host { hostname | host_ip_address }
    use-vrf vrf-name user username password password directory directory [ file file_name ]
```

## Command Default

By default, downloads the firmware to the system, reboots the system, and commits the firmware automatically. The user can specify **noactivatefirmware download** to download the firmware to the system without activating it (the node is not rebooted). The user can run **firmware activate** later to activate the firmware.

## Parameters

### coldboot

Downloads the firmware to the system and reboots both the active and standby MMs.

### directory *directory*

Specifies a fully qualified path to the directory where the firmware is located.

### file *file\_name*

Specifies the firmware .plist file. This parameter is optional; if unspecified, the default file, release.plist, is used.

### host

Specifies the host by DNS name or IP address.

#### *hostname*

Specifies an IPv4 DNS host name.

#### *host\_ip\_address*

Specifies the host IP address. IPv4 and IPv6 addresses are supported.

### manual

Updates a single management module in a chassis with two management modules. You must log in to the management module through its dedicated management IP address. This parameter is ignored when issued on a compact switch or in a chassis with only one management module.

### noactivate

Performs a firmware download without activation on the local switch.

### nocommit

Disables auto-commit mode. When auto-commit mode is disabled, firmware is downloaded only to the primary partition. You must execute the firmware **commit** command manually to propagate the new image to the secondary partition.

### noreboot

Disables auto-reboot mode. When auto-reboot mode is disabled, you must reboot the switch manually. If auto-commit mode was disabled, you must perform a manual firmware **commit** operation after the switch comes back up.

- password** *password*  
Specifies the account password.
- use-vrf** *vrf-name*  
Specifies a VRF.
- user** *username*  
Specifies the user login name for the host.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to download firmware from an external host.

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config.

By default, if you enter the firmware download command without any options, the command invokes ISSU to upgrade the entire system. ISSU involves an High Availability failover of the active management module and is non-disruptive. In contrast, both of the coldboot and default-config options involve system reboots and are disruptive to traffic.

In addition, default-config causes the loss of configuration because it resets the configuration back to the default settings during the firmware upgrade process.

If the **firmware download** command is interrupted because of an unexpected reboot, such as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before beginning another firmware download.

This command does not support pagination.

## Examples

This example downloads firmware by means of FTP and specifies a path to the directory where the firmware is located. A user login name is specified for the host and an account password is specified.

```
switch# firmware download ftp directory /buildsjc/sre/SQA/nos/nos6.0.0/nos6.0.0 host 10.31.2.27 user
releaseuser password releaseuser
```

## History

Release version	Command history
7.0.0	This command entry was enhanced.

# firmware download interactive

Allows the user to select firmware download parameters interactively before starting a firmware download.

## Syntax

```
firmware download interactive
```

## Command Default

By default, **firmware download** downloads the firmware to the system, reboots the system, and commits the firmware automatically.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to download firmware from an external host or from an attached USB device. You can run this command interactively or provide the parameters on the command line.

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config.

By default, if you enter the firmware download command without any options, the command invokes ISSU to upgrade the entire system. ISSU involves an High Availability failover of the active management module and is non-disruptive. In contrast, both of the coldboot and default-config options involve system reboots and are disruptive to traffic.

In addition, default-config causes the loss of configuration because it resets the configuration back to the default settings during the firmware upgrade process.

If the **firmware download** command is interrupted because of an unexpected reboot, such as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before beginning another firmware download.

This command does not support pagination.

## Examples

To perform a firmware download in interactive mode using default parameters:

```
device# firmware download interactive
Download to multiple nodes in the cluster? [n]:
Server name or IP address: 10.70.4.106
File name: dist
Protocol (ftp, scp, sftp, tftp) [ftp]: scp
User: fvt
Password: *****
Enter VRF name[mgmt-vrf]:
Select procedure (1=ISSU, 2=coldboot, 3=default-config) [1]:1

Performing system sanity check...

This command will cause a cold/disruptive reboot and will require that existing telnet, secure telnet
or SSH sessions be restarted.

Do you want to continue? [y/n]y
```

## History

Release version	Command history
7.0.0	This command entry was enhanced.

## Related Commands

[firmware commit](#), [firmware download](#), [show firmwaredownloadstatus](#)



# firmware download logical-chassis

Downloads the firmware onto the switches.

## Syntax

```
firmware download logical-chassis default-config { ftp | scp | sftp } host host_ip user username password password
  directory path [ rbridge-id { rbridge-id | all } ] [ auto-activate ] [ use-vrf vrf-name ]
```

## Command Default

After firmware is downloaded to the nodes, the command returns without rebooting the nodes. You will need to run **firmware activate** to activate the new firmware on the nodes.

## Parameters

### default-config

Sets the configuration back to default except the following parameters: VCS ID and RBridge ID. These three parameters are retained.

### ftp

Specifies FTP as the protocol used to download the firmware.

### scp

Specifies SCP as the protocol used to download the firmware.

### sftp

Specifies SFTP as the protocol used to download the firmware.

### host *host\_ip*

Specifies the host IP address.

### user *username*

Specifies the username.

### password *password*

Specifies the password.

### directory *path*

Specifies the filename path where the firmware is located.

### auto-activate

Specifies to automatically activate the firmware on the switches after installing the firmware.

### coldboot

Downloads the firmware to the system and reboots both the active and standby MMs.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

**use-vrf** *vrf-id*

Use this option to specify the name of the VRF where the host is located. If this option is not set, mgmt-vrf is used by default.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is controlled by a principal node (coordinator) and executed on individual nodes in the VCS cluster. All sanity check error and warning messages are displayed on the coordinator user session. If an error results during the Installation, the firmware download request is aborted.

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config. By default, if you enter the **firmware download logical-chassis** command without any options, the command invokes ISSU to upgrade the specified nodes. If you want to perform ISSU in a single step, you must specify the auto-activate option. Otherwise, this command will only download firmware to the nodes, but it won't activate the new firmware.

## Examples

This example downloads the firmware onto the switches and specifies that the firmware is automatically activated after the installation is complete.

```
device# firmware download logical-chassis ftp host 10.1.2.30 user fvt password extreme directory /
dist/nos/5.0.0bld26 file release.plist rbridge-id 1,2-3 auto-activate
Rbridge-id      Sanity Result      Current Version
-----
```

```
  1      Non-disruptive (ISSU)      5.0.0
  2      Non-disruptive (ISSU)      5.0.0
  3      Non-disruptive (ISSU)      5.0.0
```

This command will download the new firmware to the specified nodes, and cause WARM reboot on these nodes automatically.  
Do you want to continue? [y/n]: y

This example downloads the firmware to the system and reboots both the active and standby MMs.

```
device# firmware download logical-chassis ftp host 10.1.2.30 user fvt password extreme directory /
dist/nos/5.0.0bld26 file release.plist rbridge-id 1,2-3 coldboot
Rbridge-id      Sanity Result      Current Version
-----
```

```
  1      Disruptive      5.0.0
  2      Disruptive      5.0.0
  3      Disruptive      5.0.0
```

This command will download the new firmware to the specified nodes, and reboot them automatically.  
Do you want to continue? [y/n]: y

This example downloads the firmware to the system and resets the default configuration with the exception of the VCS ID and RBridge ID parameters.

```
device# firmware download logical-chassis default-config ftp host 10.1.2.30 user fvt password extreme
directory /dist/nos/5.0.0bld26 file release.plist rbridge-id 1,2-3
Rbridge-id      Sanity Result      Current Version
-----
1      Disruptive      5.0.0
2      Disruptive      5.0.0
3      Disruptive      5.0.0
You are invoking firmware download with the provision option. This command will
download the new firmware to the specified nodes, default their configuration, and
reboot them automatically.
Do you want to continue? [y/n]: y
```

## History

Release version	Command history
7.0.0	This command entry was enhanced.

## Related Commands

[firmware activate](#), [firmware recover](#), [show firmwaredownloadstatus](#)

# firmware download scp

Specifies Secure Copy (SCP) as the protocol used to perform a firmware download.

## Syntax

```
firmware download scp [ coldboot ] [ manual ] [ nocommit ] [ noreboot ] host { hostname | host_ip_address } user username
password password directory directory [ file file_name ] [ noactivate ] [ use-vrf vrf-name]
```

## Command Default

A filename is optional. If no filename is specified, release.plist, is used.

## Parameters

### coldboot

Downloads the firmware to the system and reboots both the active and standby MMs.

### manual

Performs a firmware download on the local switch.

### nocommit

Disables auto-commit mode. When auto-commit mode is disabled, firmware is downloaded only to the primary partition. You must execute the firmware **commit** command manually to propagate the new image to the secondary partition. (Skips auto-commit after firmware download.)

### noreboot

Disables auto-reboot mode. When auto-reboot mode is disabled, you must reboot the switch manually. If auto-commit mode was disabled, you must perform a manual firmware **commit** operation after the switch comes back up.

### host

Specifies the host by DNS name or IP address.

#### hostname

Specifies an IPv4 DNS host name.

#### host\_ip\_address

Specifies the host IP address. IPv4 and IPv6 addresses are supported.

### user username

Specifies the user login name for the host.

### password password

Specifies the account password.

### directory directory

Specifies a fully qualified path to the directory where the firmware is located.

### file file\_name

Specifies the firmware .plist file. This parameter is optional.

### noactivate

Performs a firmware download without activation on the local switch.

**use-vrf** *vrf-id*

Use this option to specify the name of the VRF where the host is located. If this option is not set, mgmt-vrf is used by default.

## Modes

Privileged EXEC mode.

## Usage Guidelines

Use this command to download firmware from an external host or from an attached USB device. You can run this command interactively or provide the parameters on the command line.

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config.

By default, if you enter the firmware download command without any options, the command invokes ISSU to upgrade the entire system. ISSU involves an High Availability failover of the active management module and is non-disruptive. In contrast, both of the coldboot and default-config options involve system reboots and are disruptive to traffic.

In addition, default-config causes the loss of configuration because it resets the configuration back to the default settings during the firmware upgrade process.

If the **firmware download** command is interrupted because of an unexpected reboot, such as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before beginning another firmware download.

This command does not support pagination.

If the **firmware download** is interrupted because of an unexpected reboot as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before starting another firmware download.

## Examples

This example downloads firmware by means of SCP and specifies a path to the directory where the firmware is located. A user login name is specified for the host and an account password is specified.

```
switch# firmware download scp directory /buildsjc/sre/SQA/nos/nos6.0.0/nos6.0.0 host 10.31.2.27 user
releaseuser password releaseuser
```

## History

Release version	Command history
7.0.0	This command entry was enhanced.

## Related Commands

[firmware commit](#), [firmware restore](#), [show firmwaredownloadstatus](#)

# firmware download sftp

Specifies Secure FTP (SFTP) as the protocol used to perform a firmware download.

## Syntax

```
firmware download sftp [ coldboot ] directory directory [ manual ] [ nocommit ] [ noreboot ] host { hostname |
host_ip_address } user username password password directory directory [ file file_name ] [ noactivate ] [ use-vrf vrf-
name ]
```

## Parameters

### **coldboot**

Downloads the firmware to the system and reboots both the active and standby MMs.

### **directory** *directory*

Specifies a fully qualified path to the directory where the firmware is located.

### **file** *filename*

Specifies the firmware .plist file. This parameter is optional; if unspecified, the default file, release.plist, is used.

### **host**

Specifies the host by DNS name or IP address.

#### *hostname*

Specifies an IPv4 DNS host name.

#### *host\_ip\_address*

Specifies the host IP address. IPv4 and IPv6 addresses are supported.

### **manual**

Performs a firmware download on the local switch.

### **noactivate**

Performs a firmware download without activation on the local switch.

### **nocommit**

Disables auto-commit mode. When auto-commit mode is disabled, firmware is downloaded only to the primary partition. You must execute the firmware **commit** command manually to propagate the new image to the secondary partition. (Skips auto-commit after firmware download.)

### **noreboot**

Disables auto-reboot mode. When auto-reboot mode is disabled, you must reboot the switch manually. If auto-commit mode was disabled, you must perform a manual firmware **commit** operation after the switch comes back up.

### **password** *password*

Specifies the account password.

### **user** *username*

Specifies the user login name for the host.

### **use-vrf** *vrf-id*

Use this option to specify the name of the VRF where the host is located. If this option is not set, mgmt-vrf is used by default.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to download firmware from an external host or from an attached USB device. You can run this command interactively or provide the parameters on the command line.

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config.

By default, if you enter the firmware download command without any options, the command invokes ISSU to upgrade the entire system. ISSU involves an High Availability failover of the active management module and is non-disruptive. In contrast, both of the coldboot and default-config options involve system reboots and are disruptive to traffic.

In addition, default-config causes the loss of configuration because it resets the configuration back to the default settings during the firmware upgrade process.

If the **firmware download** command is interrupted because of an unexpected reboot, such as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before beginning another firmware download.

This command does not support pagination.

If the **firmware download** is interrupted because of an unexpected reboot as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before starting another firmware download.

## Examples

This example downloads firmware by means of SFTP and specifies a path to the directory where the firmware is located. A user login name is specified for the host and an account password is specified.

```
switch# firmware download sftp directory /buildsjc/sre/SQA/nos/nos6.0.0/nos6.0.0 host 10.31.2.27 user
releaseuser password releaseuser
```

## History

Release version	Command history
7.0.0	This command entry was enhanced.

# firmware download tftp

Specifies Trivial FTP (TFTP) as the protocol used to perform a firmware download.

## Syntax

```
firmware download tftp [ coldboot ] directory directory [ manual ] [ nocommit ] [ noreboot ] host { hostname |
host_ip_address } user username password password directory directory [ file file_name ] [ noactivate ] [ use-vrf vrf-
name ]
```

## Parameters

### **coldboot**

Downloads the firmware to the system and reboots both the active and standby MMs.

### **directory** *directory*

Specifies a fully qualified path to the directory where the firmware is located.

### **file** *filename*

Specifies the firmware .plist file. This parameter is optional; if unspecified, the default file, release.plist, is used.

### **host**

Specifies the host by DNS name or IP address.

#### *hostname*

Specifies an IPv4 DNS host name.

#### *host\_ip\_address*

Specifies the host IP address. IPv4 and IPv6 addresses are supported.

### **manual**

Performs a firmware download on the local switch.

### **noactivate**

Performs a firmware download without activation on the local switch.

### **nocommit**

Disables auto-commit mode. When auto-commit mode is disabled, firmware is downloaded only to the primary partition. You must execute the firmware **commit** command manually to propagate the new image to the secondary partition. (Skips auto-commit after firmware download.)

### **noreboot**

Disables auto-reboot mode. When auto-reboot mode is disabled, you must reboot the switch manually. If auto-commit mode was disabled, you must perform a manual firmware **commit** operation after the switch comes back up.

### **password** *password*

Specifies the account password.

### **user** *username*

Specifies the user login name for the host.

### **use-vrf** *vrf-id*

Use this option to specify the name of the VRF where the host is located. If this option is not set, mgmt-vrf is used by default.



## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to download firmware from an external host or from an attached USB device. You can run this command interactively or provide the parameters on the command line.

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config.

By default, if you enter the firmware download command without any options, the command invokes ISSU to upgrade the entire system. ISSU involves an High Availability failover of the active management module and is non-disruptive. In contrast, both of the coldboot and default-config options involve system reboots and are disruptive to traffic.

In addition, default-config causes the loss of configuration because it resets the configuration back to the default settings during the firmware upgrade process.

If the **firmware download** command is interrupted because of an unexpected reboot, such as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before beginning another firmware download.

This command does not support pagination.

If the **firmware download** is interrupted because of an unexpected reboot as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before starting another firmware download.

## Examples

This example downloads firmware by means of TFTP and specifies a path to the directory where the firmware is located. The host is specified by IP address and a firmware .plist file is specified.

```
switch# firmware download tftp directory /buildsjc/sre/SQA/nos/nos6.0.0/nos6.0.0 host 10.31.2.27 file
release.plist
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	This command was enhanced.

# firmware download usb

Specifies USB as the protocol used to perform a firmware download.

## Syntax

```
firmware download usb [ coldboot ] [ noactivate ] [ nocommit ] [ noreboot ] [ manual ] directory directory
```

## Command Default

By default, the **firmware download** process reboots the system and activates the new image. Finally, the process performs a **firmware commit** operation to copy the new image to the other partition.

## Parameters

### coldboot

Downloads the firmware to the system and reboots both the active and standby MMs. **Caution:** Do not use this option unless instructed to do so by Extreme Technical Support.

### directory *directory*

Specifies a fully qualified path to the directory where the firmware is located.

### manual

Updates a single management module in a chassis with two management modules. You must log in to the management module through its dedicated management IP address. This parameter is ignored when issued on a Top-of-Rack (ToR) switch or in a chassis with only one management module.

### noactivate

Performs a firmware download without activation on the local switch.

### nocommit

Disables auto-commit mode. When auto-commit mode is disabled, firmware is downloaded only to the primary partition. You must execute the firmware **commit** command manually to propagate the new image to the secondary partition.

### noreboot

Disables auto-reboot mode. When auto-reboot mode is disabled, you must reboot the switch manually. If auto-commit mode was disabled, you must perform a manual firmware **commit** operation after the switch comes back up.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to download firmware from an external host or from an attached USB device. You can run this command interactively or provide the parameters on the command line.

You can use one of the following options for firmware upgrade/downgrade; ISSU, coldboot, or default-config.

By default, if you enter the firmware download command without any options, the command invokes ISSU to upgrade the entire system. ISSU involves an High Availability failover of the active management module and is non-disruptive. In contrast, both of the coldboot and default-config options involve system reboots and are disruptive to traffic.

In addition, default-config causes the loss of configuration because it resets the configuration back to the default settings during the firmware upgrade process.

If the **firmware download** command is interrupted because of an unexpected reboot, such as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before beginning another firmware download.

This command does not support pagination.

If the **firmware download** is interrupted because of an unexpected reboot as a result of a software error or power failure, the command automatically recovers the corrupted secondary partition. Wait for the recovery to complete before starting another firmware download.

## Examples

To download firmware from an attached USB device using the command line:

```
switch# firmware download usb directory NOS_v5.0.0
```

## History

Release version	Command history
7.0.0	This command entry was enhanced.

## Related Commands

[firmware commit](#), [firmware restore](#), [show firmwaredownloadstatus](#)

# firmware install

Installs new software but deletes all configuration in the system.

## Parameters

### ftp

Specifies FTP as the protocol used to install the firmware.

### scp

Specifies SCP as the protocol used to install the firmware.

### host

Specifies the host by DNS name or IP address.

*host\_ip\_address*

Specifies the host IP address. IPv4 and IPv6 addresses are supported.

### use-vrf *vrf-name*

Specifies a VRF.

### user *username*

Specifies the user login name for the host.

### password *password*

Specifies the account password.

### directory *directory*

Specifies a fully qualified path to the directory where the firmware is located.

### file *filename*

Specifies the firmware .plist file. This parameter is optional; if unspecified, the default file, release.plist, is used.

### [ **vcs-mode** { **1** | **2** }

Specifies the VCS mode for the switch when the new firmware has been installed. If not set, the platform-dependent default VCS mode is used. Values you can select are: **1** (logical chassis cluster mode), or **2** (standalone mode).

### **vcs-id** *vcs\_id*

Specifies the new VCS ID when the new firmware has been installed. If not set, the platform-dependent default VCS ID is used. Range is 1 to 8192.

### **rbridge-id** *rbridge-id*

Specifies the new RBridge ID when the new firmware has been installed. If not set, the platform-dependent default RBridge ID is used. Range is 1 to 239.

### **manual**

In a dual-MM system, if the manual option is used, after firmware install is completed on both MMs, reboot them at the same time. Otherwise, the MMs may run into configuration mismatch issue.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command cleans the existing firmware on the system before installing the new firmware. All configurations in the system is completely lost.

By default, **firmware install** installs the firmware on both the active and standby modules. If the manual option is specified, then the firmware is installed on the local module only.



### CAUTION

Do not use this command unless instructed by Extreme Technical Support.

## Examples

To install new firmware, delete all existing configurations, and to specify the vcs mode, vcs ID and RBridge ID you want the switch to come up in:

```
switch# firmware install scp host 10.70.4.109 user fvt pass pray4green directory /buildsjc/sre/SQA/nos/nos5.0.0/nos5.0.0_bld26 vcs-mode 1 vcs-id 6 rbridgeid 10
```

# firmware peripheral-update microcode

Installs firmware (microcode) on PHY chips on an Extreme VDX 6740T or VDX 6740T-1G switch from a given binary image.

## Syntax

```
firmware peripheral-update microcode phy { flash | usb } string
```

## Command Default

The peripheral firmware has not been updated.

## Parameters

### flash

Designates that the source file is on a flash drive. When upgrading the AQ1402 PHY from flash, the firmware has been copied to the flash:// folder on the device.

### usb

Designates that the source file is on an Extreme USB device. When Extreme USB is used to update AQ1402 PHY, then the PHY microcode firmware must be resident in the firmware\ directory on the USB.

### string

The name of the source file used for the update. The string can not be more than 128 bytes long.

## Modes

Privileged EXEC mode

## Usage Guidelines

When upgrading the AQ1402 PHY from flash, it is expected that the firmware has been copied to the flash:// folder.

### NOTE

If you are planning to power-cycle the device after a microcode update, it is recommended that you use the **chassis power-cycle-db-shutdown** command before power-cycling the device.

## Examples

Example of typical execution using a USB source file.

```
device# firmware peripheral-update microcode phy usb Firmware_1.38.c1_Brocade.Castor.cld
Starting to Program AQ Flash.....
AQ1402[0xbb300000] phy handle
CRC check good on image file
Taking Control Of the Flash Interface..
Determining Flash Type...
bootLoad=0x2
primary=0x2
FLASH type = Atmel AT45DB041D
Erasing Flash.....
Atmel AT45DB041D Erase Started
Writing Image to Flash .....
Bytes in file 0x2E000

Starting read of FLASH data
OK
AQ Flash Upgrade Successful
2016/07/19-05:51:06, [SULB-1000], 13874, SW/0 | Active, WARNING, VDX6740T-1G, The firmware download
command has been started.
AQ Firmware_1.38.c1_Extreme.Castor.cld upgrade success...PowerCycle/Reboot Switch.

device# reload
Are you sure you want to reload the switch? [y/n]:y
The system is going down for reload NOW !!
```

Example of typical execution using a flash source file.

```
device# firmware peripheral-update microcode phy flash Firmware_1.38.c1_Brocade.Castor.cld
Starting to Program AQ Flash.....
AQ1402[0xbb300000] phy handle
CRC check good on image file
Taking Control Of the Flash Interface..
Determining Flash Type...
bootLoad=0x2
primary=0x2
FLASH type = Atmel AT45DB041D
Erasing Flash.....
Atmel AT45DB041D Erase Started
Writing Image to Flash .....
Bytes in file 0x2E000

Starting read of FLASH data
OK
AQ Flash Upgrade Successful
2016/07/19-05:53:24, [SULB-1000], 13875, SW/0 | Active, WARNING, VDX6740T-1G, The firmware download
command has been started.
AQ Firmware_1.38.c1_Brocade.Castor.cld upgrade success...PowerCycle/Reboot Switch.

device# reload
Are you sure you want to reload the switch? [y/n]:y
The system is going down for reload NOW !!
```

## History

Release version	Command history
7.1.0	This command was introduced.

## Related Commands

[chassis power-cycle-db-shutdown](#), [reload](#), [show version](#)

# firmware recover

Recovers the previous firmware version on the switch nodes if a firmware upgrade was unsuccessful.

## Syntax

```
firmware recover [ rbridge-id { rbridge-id | all }]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command undoes the operation that was performed using the firmware download "noactivate" option.

If you invoke a noactivate firmware download, the firmware is loaded to the secondary node without swapping partitions. If firmware recover is executed, it will perform a forceful commit in that node. This CLI does not reboot the node.

## Examples

To recover firmware on switch nodes 1, 2, 3, and 5:

```
device# firmware recover rbridge-id rid1-rid3,rid5
```

To perform a firmware recovery on all switch nodes:

```
device# firmware recover rbridge-id all
```

## Related Commands

[firmware activate](#), [show firmwaredownloadstatus](#)



# firmware restore

Swaps the partition and reboots the node.

## Syntax

`firmware restore`

## Modes

Privileged EXEC mode

## Usage Guidelines



### CAUTION

Do not use this command unless instructed by Extreme Technical Support.

Use this command to restore the previously active firmware image. You can run this command only if auto-commit mode was disabled during the firmware download. After a firmware download and a reboot (with auto-commit mode disabled), the downloaded firmware becomes active. If you do not want to commit the firmware, use the **firmware restore** command.

This command reboots the system and reactivates the previous firmware. After reboot, all primary and secondary partitions restore the previous firmware image.

This command causes the node to boot up with its older firmware. Later, the image in the primary partition is automatically committed to the secondary partition.

This command is supported only on the local management module.

The **firmware download** command must have been run with the **nocommit** parameter for the **firmware restore** operation to succeed.

## Examples

To restore the previous firmware:

```
switch# firmware restore

Restore old image to be active ...
Restore both primary and secondary image after reboot.
The system is going down for reboot NOW !!
Broadcast message from root (ttyS0) Fri Oct 26 23:48:54 2001...
Doing firmwarecommit now.
Please wait ...
```

## Related Commands

[firmware commit](#), [firmware download](#)

# firmware sync

Synchronizes the firmware on the current switch to the standby Management Module (MM).

## Syntax

**firmware sync**

## Modes

Privileged EXEC mode

## Examples

To install new firmware, delete all existing configurations, and to specify the vcs mode, vcs ID and RBridge ID you want the switch to come up in:

```
switch# firmware sync
```

This command will approximately take 15 minutes to complete.

It will cause the standby MM to reboot during the process. All CLIs on active MM in this login session will be blocked until the process is complete.

```
Do you want to continue? [y/n]:y
```

# flexport

Used in conjunction with the **hardware** command to set the switch into FlexPort configuration mode.

## Syntax

```
flexport rbridge-id/slot/port[:breakout-index]
```

## Parameters

*rbridge-id/slot/port*

Specifies a valid Fibre Channel port interface.

*rbridge-id*

Specifies the RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies which port to modify.

*:breakout-index*

Specifies which port to modify.

## Usage Guidelines

Refer to [hardware](#) on page 607.

## Modes

Hardware configuration mode

## Examples

This command configures port 5 on Rbridge ID 1:

```
switch# configure terminal
switch(config)# hardware
switch(config-hw)# flexport 1/0/5
switch(conf-hw-flex-1/0/5)# type fibrechannel
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[connector-group](#), [hardware](#), [speed \(FlexPort\)](#)

# forward-delay

Specifies the time an interface spends in each of the listening and learning states.

## Syntax

**forward-delay** *seconds*  
**no forward-delay**

## Command Default

15 seconds

## Parameters

*seconds*

Specifies the time that an interface spends in the Spanning Tree Protocol (STP) learning and listening states. Valid values range from 4 through 30 seconds.

## Modes

Spanning tree configuration mode

## Usage Guidelines

This command specifies how long the listening and learning states last before the interface begins the forwarding of all spanning-tree instances.

STP interface states:

- Listening - The interface processes the Bridge Protocol Data Units (BPDUs) and awaits possible new information that might cause it to return to the blocking state.
- Learning - The interface does not yet forward frames (packets), instead it learns source addresses from frames received and adds them to the filtering database (switching database).
- Forwarding - An interface receiving and sending data, normal operation. STP still monitors incoming BPDUs that can indicate it should return to the blocking state to prevent a loop.
- Blocking - An interface that can cause a switching loop, no user data is sent or received, but it might go to the forwarding state if the other links in use fail and the STP determines that the interface may transition to the forwarding state. BPDU data continues to be received in the blocking state.

When you change the spanning-tree forward-delay time, it affects all spanning-tree instances. When configuring the forward-delay, the following relationship should be kept:

$$(2 \times (\text{forward-delay} - 1)) \geq \text{max-age} \geq (2 \times (\text{hello-time} + 1))$$

Enter **no forward-delay** to return to the default settings.

The command is the same regardless of which type of STP is enabled.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes.

## Examples

To configure the forward-delay time to 18 seconds:

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# forward-delay 18
```

```
device# configure terminal
device(config)## protocol spanning-tree rstp
device(conf-rstp)# forward-delay 18
```

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# forward-delay 18
```

```
device# configure terminal
device(config)# protocol spanning-tree pvst
device(conf-pvst)# forward-delay 18
```

```
device# configure terminal
device(config)# protocol spanning-tree rpvt
device(conf-rpvst)# forward-delay 18
```

## Related Commands

[hello-time](#), [max-age](#)

# gateway-address

Configures the gateway IP address for IPv4 or IPv6 Fabric-Virtual-Gateway sessions.

## Syntax

**gateway-address** *gateway-address*

**no gateway-address** *gateway-address*

## Command Default

None

## Parameters

*gateway-address*

IPv4 or IPv6 address in the format A.B.C.D/L or x:x:x::x/L.

## Modes

IPv4 or IPv6 Fabric-Virtual-Gateway on a VE interface configuration mode

## Usage Guidelines

Enter the **no** form of the command to remove the gateway IP address for the IPv4 or IPv6 Fabric-Virtual-Gateway from global VE configuration mode.

## Examples

The following example shows how to configure the gateway IP address for IPv4 Fabric-Virtual-Gateway on the VE interface.

```
device(config)# interface ve 2000
device(config-Ve-2000)# ip fabric-virtual-gateway
device(config-ip-fabric-virtual-gw)# gateway-address 192.128.2.1/24
```

The following example shows how to configure the gateway IP address for IPv6 Fabric-Virtual-Gateway on the VE interface.

```
device(config)# interface ve 2000
device(config-Ve-2000)# ipv6 fabric-virtual-gateway
device(config-ipv6-fabric-virtual-gw)# gateway-address 2001:1:0:1::1/64
```

## History

Release version	Command history
5.0.1	This command was introduced.

# gateway-mac-address

Configures the gateway MAC address for IPv4 or IPv6 Fabric-Virtual-Gateway sessions.

## Syntax

```
gateway-mac-address mac-address
no gateway-mac-address
```

## Command Default

Default gateway MAC address for IPv4 is **02e0.5200.01ff** and for IPv6 is **02e0.5200.02fe**.

## Parameters

*mac-address*  
Gateway MAC address in HHHH.HHHH.HHHH format.

## Modes

Address-family configuration mode

## Usage Guidelines

Before configuring or unconfiguring a user-defined gateway MAC address, the address-family must be disabled administratively. The address-family can be disabled by means of the **no enable** command under Fabric-Virtual-Gateway address-family configuration mode.

Enter the **no** form of the command to remove a gateway MAC address for the IPv4 or IPv6 Fabric-Virtual-Gateway session.

## Examples

The following example shows how to configure the gateway MAC address for an IPv4 Fabric-Virtual-Gateway session.

```
device(config)# router fabric-virtual-gateway
device(conf-router-fabric-virtual-gateway)# address-family ipv4
device(conf-address-family-ipv4)# gateway-mac-address 0011.2222.2233
```

## History

Release version	Command history
5.0.1	This command was introduced.

## Related Commands

[accept-unicast-arp-request](#)

# graceful-restart (BGP)

Enables the BGP graceful restart capability.

## Syntax

```
graceful-restart [ purge-time seconds | restart-time seconds | stale-routes-time seconds ]
no graceful-restart
```

## Command Default

Disabled.

## Parameters

### purge-time

Specifies the maximum period of time, in seconds, for which a restarting device maintains stale routes in the BGP routing table before purging them. The default value is 600 seconds. The configurable range of values is from 1 to 3600 seconds.

### restart-time

Specifies the restart-time, in seconds, advertised to graceful restart-capable neighbors. The default value is 120 seconds. The configurable range of values is from 1 to 3600 seconds.

### stale-routes-time

Specifies the maximum period of time, in seconds, that a helper device will wait for an End-of-RIB (EOR) message from a peer. All stale paths are deleted when this time period expires. The default value is 360 seconds. The configurable range of values is from 1 to 3600 seconds.

## Modes

BGP address-family IPv4 unicast configuration mode  
 BGP address-family IPv6 unicast configuration mode  
 BGP address-family IPv4 unicast VRF configuration mode  
 BGP address-family IPv6 unicast VRF configuration mode  
 BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

Use this command to enable or disable the graceful-restart capability globally for all BGP neighbors in a BGP network. When this command is enabled, graceful-restart capability is negotiated with neighbors in the BGP OPEN message when a session is established. If the neighbor advertises support for graceful restart, that function is activated for that neighbor session. Otherwise, graceful restart is not activated for that session, even though it is enabled locally. If the neighbor has not sent graceful-restart parameters, the restarting device will not wait for the neighbor to start route calculation, but graceful restart will be enabled.

If the graceful-restart capability is enabled after a BGP session has been established, the neighbor session must be cleared for graceful restart to take effect.



The **purge-time** parameter is applicable for both restarting and helper devices. The timer starts when a BGP connection is closed. The timer ends when an EOR is received from all nodes, downloaded into BGP and an EOR sent to all neighbors. The configured purge-time timer value is effective only on the configured node.

The **restart-time** parameter is applicable only for helper devices. The timer starts at the time the BGP connection is closed by the remote peer and ends when the Peer connection is established. The configured restart-time timer value is effective only on the peer node, and not in the configured node. During negotiation time, the timer value is exchanged.

The **stale-routes-time** parameter is applicable only for helper devices. The timer starts when the peer connection is established after the HA-failover. The timer ends at the time an EOR is received from the peer. The configured stale-time timer value is effective only on the configured node.

For non-default VRF instances, graceful restart timers are inherited from the default VRF.

Use the **clear ip bgp neighbor** command with the **all** parameter for the changes to the graceful-restart parameters to take effect immediately.

The **no** form of the command disables the BGP graceful-restart capability globally for all BGP neighbors.

## Examples

The following example enables the BGP graceful restart capability.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1.1.1.1 remote-as 2
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# graceful-restart
```

The following example sets the purge time to 240 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1.1.1.1 remote-as 2
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# graceful-restart
%Warning: Please clear the neighbor session for the parameter change to take effect!
device(config-bgp-ipv4u)# graceful-restart purge-time 240
```

The following example sets the restart time to 60 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1.1.1.1 remote-as 2
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# graceful-restart
%Warning: Please clear the neighbor session for the parameter change to take effect!
device(config-bgp-ipv4u)# graceful-restart restart-time 60
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

The following example sets the stale-routes time to 180 seconds.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1000::1 remote-as 2
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 1000::1 activate
device(config-bgp-ipv6u)# graceful-restart
%Warning: Please clear the neighbor session for the parameter change to take effect!
device(config-bgp-ipv6u)# graceful-restart stale-routes-time 180
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

The following example enables the BGP graceful restart capability and sets the purge time to 220 seconds in L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# graceful-restart purge-time 220
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.

## Related Commands

[address-family l2vpn evpn](#), [address-family unicast \(BGP\)](#), [router bgp](#)

# graceful-restart (OSPFv2)

Enables the OSPF Graceful Restart (GR) capability.

## Syntax

```
graceful-restart [ helper-disable | restart-time seconds ]
no graceful-restart
```

## Command Default

Graceful restart and graceful restart helper capabilities are enabled.

## Parameters

### helper-disable

Disables the GR helper capability.

### restart-time

Specifies the maximum restart wait time, in seconds, advertised to neighbors. The default value is 120 seconds. The configurable range of values is from 10 through 1800 seconds.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use **no graceful-restart helper-disable** to re-enable the GR helper capability.

The **no** form of the command disables the graceful restart capability.

## Examples

The following example disables the GR capability.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router ospf
device(config-router-ospf-vrf-default-vrf)# no graceful-restart
```

The following example disables the GR helper capability.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router ospf
device(config-router-ospf-vrf-default-vrf)# graceful-restart helper-disable
```

The following example re-enables the GR helper capability.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router ospf
device(config-router-ospf-vrf-default-vrf)# no graceful-restart helper-disable
```

The following example re-enables the GR capability and changes the maximum restart wait time from the default value to 240 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router ospf
device(config-router-ospf-vrf-default-vrf)# graceful-restart restart-time 240
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[graceful-restart helper \(OSPFv3\)](#), [nonstop-routing \(OSPF\)](#)

# graceful-restart helper (OSPFv3)

Enables the OSPFv3 graceful restart (GR) helper capability.

## Syntax

```
graceful-restart helper { disable | strict-lsa-checking }
no graceful-restart helper
```

## Command Default

GR helper is enabled.

## Parameters

### disable

Disables the OSPFv3 GR helper capability.

### strict-lsa-checking

Enables the OSPFv3 GR helper mode with strict link-state advertisement (LSA) checking.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command disables the GR helper capability on a device.

## Examples

The following example enables GR helper and sets strict LSA checking.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# graceful-restart helper strict-lsa-checking
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[graceful-restart \(OSPFv2\)](#), [nonstop-routing \(OSPF\)](#)

# graceful-shutdown

Gracefully shuts down all BGP neighbors.

## Syntax

```
graceful-shutdown seconds [ community value [ local-preference value ] ] | local-preference value [ community value ] |
route-map route-map-name ]
```

```
no graceful-shutdown seconds [ community value [ local-preference value ] ] | local-preference value [ community value ] |
route-map route-map-name ]
```

## Command Default

Default graceful shutdown parameters are applied.

## Parameters

*seconds*

Specifies the number of seconds in which the BGP graceful shutdown will occur. Valid values range from 30 through 600 seconds.

**community** *value*

Sets the community attribute for graceful shutdown. Valid values range from 1 through 4294967295.

**local-preference** *value*

Sets the local preference attribute for graceful shutdown. Valid values range from 0 through 4294967295.

**route-map** *route-map-name*

Specifies the route map for graceful shutdown attributes.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command de-activates graceful shutdown.

## Examples

The following example gracefully shuts down all BGP neighbors and sets the graceful shutdown timer to 180 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# graceful-shutdown 180
```

The following example gracefully shuts down all BGP neighbors and sets the graceful shutdown timer to 600 seconds. The route map "myroutemap" is specified for graceful shutdown attributes.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# graceful-shutdown 600 route-map myroutemap
```

The following example gracefully shuts down all BGP neighbors and sets the graceful shutdown timer to 600 seconds. The community attribute is set to 10.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# graceful-shutdown 600 community 10
```

## History

Release version	Command history
7.2.0	This command was introduced.

# gratuitous-arp timer

Configures the global gratuitous ARP timer in VCS.

## Syntax

```
gratuitous-arp timer value
no gratuitous-arp timer
```

## Command Default

The gratuitous ARP timer is disabled.

## Parameters

*value*

Gratuitous ARP timer in seconds. The range is from 0 through 360.

## Modes

Fabric-Virtual-Gateway address-family IPv4 or IPv6 configuration mode

Fabric-Virtual-Gateway global interface VE IPv4 or IPv6 configuration mode

## Usage Guidelines

Enter the **no** form of the command to remove the configured gratuitous ARP timer value.

## Examples

The following example shows how to configure the gratuitous ARP timer in Fabric-Virtual-Gateway address-family IPv4 configuration mode.

```
device(config)# router fabric-virtual-gateway
device(conf-router-fabric-virtual-gateway)# address-family ipv4
device(conf-address-family-ipv4)# gratuitous-arp timer 15
```

The following example shows how to configure the gratuitous ARP timer in Fabric-Virtual-Gateway under VE interface IPv4 configuration mode.

```
device(config)# interface ve 2000
device(config-ve-2000)# ip fabric-virtual-gateway
device(config-ip-fabric-virtual-gw)# gratuitous-arp timer 15
```

The following example shows how to configure the gratuitous ARP timer in Fabric-Virtual-Gateway address-family IPv6 configuration mode.

```
device(config)# router fabric-virtual-gateway
device(conf-router-fabric-virtual-gateway)# address-family ipv6
device(conf-address-family-ipv6)# gratuitous-arp timer 60
```



The following example shows how to configure the gratuitous ARP timer in Fabric-Virtual-Gateway under VE interface IPv6 configuration mode.

```
device(config)# interface ve 3000
device(config-Ve-3000)# ipv6 fabric-virtual-gateway
device(config-ipv6-fabric-virtual-gw)# gratuitous-arp timer 60
```

## History

Release version	Command history
5.0.1	This command was introduced.

# group

Creates a user-defined logical group for either SFP or Ethernet ports for use in Monitoring and Alerting Priority Suite (MAPS).

## Syntax

```
group { name type { sfp | interface } members member_list }  
no group name
```

## Command Default

No groups exist by default.

## Parameters

*name*

Descriptive name for the logical group.

**type**

Defines which type of port is assigned to the members of the group.

**sfp**

Configures the logical group as SFP ports.

**interface**

Configures the logical group as Ethernet ports.

*members member\_list*

Defines the members of the group. Members are either Ethernet interfaces or SFPs, separated by commas.

## Modes

MAPS configuration mode

## Usage Guidelines

Use the **no group *name*** command to remove the group.

## Examples

Typical command example:

```
device#configure terminal  
device(config)# rbridge-id 1  
device(config-rbridge-id-1)# maps  
device(config-rbridge-id-1-maps)# group EthUser type interface members 1/0/1,1/0/2,1/0/3
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[policy, rule \(MAPS\)](#)

# ha chassisreboot

Performs a reboot of the chassis.

## Syntax

**ha chassisreboot**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command reboots the entire chassis. Both the active and the standby management module reboot. Both management modules retain their original high-availability (HA) role after the system comes back up. If the power-on self test (POST) is enabled, it is executed when the system comes back up.

This command is supported only on the active management module. This command is not supported on the standby management module. Both management modules must be in sync for the HA reboot operation to succeed. Failover and reboots can be disruptive.

## Examples

To perform an HA reboot:

```
switch# ha chassisreboot
```

# ha disable

Disables the High Availability (HA) feature on a device.

## Syntax

**ha disable**

## Command Default

HA is disabled.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the HA feature is already disabled, this command has no effect.

This command is supported only on the active management module.

This command is not supported on the standby management module.

Do not use this command unless instructed by Extreme Technical Support.

### NOTE

With the introduction of Network OS 4.0, failover is no longer disruptive for Layer 2.

## Examples

To disable HA on a device:

```
switch# ha disable
```

```
Service instances out of sync  
1970/01/01-00:06:30, [HASM-1101], 111, MM1, WARNING, chassis, HA State out of sync.  
HA is disabled
```

## Related Commands

[ha enable](#), [reload](#), [show ha](#)

# ha enable

Enables the High Availability (HA) feature on a device.

## Syntax

**ha enable**

## Command Default

HA is disabled.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the HA feature is already enabled, this command has no effect. If the HA feature is disabled, this command enables it. The standby management process reboots as part of the process.

Running the command displays a warning message and prompts for confirmation before rebooting the management module.

You can also use this command to display the servers that are running the syslogd daemon and those that system messages are sent to. Servers are specified in the configuration database by IP address.

This command is supported only on the local management module.

This command is not supported on the standby management module.

Do not use this command unless instructed by Extreme Technical Support.

### NOTE

With the introduction of Network OS 4.0, failover is no longer disruptive for all Layer 2.

## Examples

To enable HA on a device:

```
switch# ha enable
```

```
Warning: This command will enable the HA. It will reboot the standby CP and
require all telnet, secure telnet, and SSH sessions to the standby CP to be
restarted.
```

```
Are you sure you want to go ahead [y/n]? y
```

```
1970/01/01-00:07:18, [EM-1047], 113, MM1, INFO, chassis, CP in slot 2 not faulty, CP ERROR deasserted.
HTBT hit a threshold: 2
HTBT hit a threshold: 2
Heartbeat to 2 Down!
  resetting peer
  resetting peer 127.2.1.2
HA is enabled
```

## Related Commands

ha disable, ha failover, reload, show ha

# ha failover

Initiates a failover from the active to the standby management module (MM).

## Syntax

`ha failover`

## Modes

Privileged EXEC mode

## Usage Guidelines

This command forces a failover from the active to the standby MM. When the process completes, the former standby performs a warm recovery and takes over as the active MM. If the active and standby MMs are not synchronized, the command aborts.

## Examples

To perform a failover:

```
switch# ha failover
```



# ha sync start

This command is used to enable the high availability (HA) state synchronization.

## Syntax

```
ha sync start
```

## Modes

Privileged EXEC mode

## Examples

To enable HA synchronization:

```
switch# ha sync start
Are you sure you want to start sync [y/n]
All service instances in sync
2012/10/06-16:10:36, [HASM-1100], 630, M2, INFO, VDX8770-4, HA State is in sync.
```

# ha sync stop

Disables high availability state synchronization on a device.

## Syntax

**ha sync stop**

## Command Default

Synchronization is enabled.

## Modes

Privileged EXEC mode

## Examples

To disable state synchronization:

```
switch# ha sync stop
Are you sure you want to stop sync [y/n]? y
Service instances out of sync
2012/10/06-16:06:13, [HASM-1101], 619, M2, WARNING, VDX8770-4, HA State out of sync.
```

## Related Commands

[ha failover](#), [ha enable](#), [ha sync start](#)

# hardware

Enters hardware configuration mode to enter a variety of configuration modes.

## Syntax

**hardware**

## Modes

Global configuration mode

## Usage Guidelines

Use this command to enter hardware configuration mode and configure the following options:

- Use the **connector** command to configure QSFP breakout mode.
- Use the **connector-group** command to designate which connector group a FlexPort-enabled interface is allowed to access on a switch.
- Use the **custom-profile** command to specify a custom hardware profile.
- Use the **flexport** command to enable FlexPort capability on Fibre Channel or Ethernet interfaces.
- Use the **port-group** command to configure port group performance or density mode on Extreme 27x40 GbE line cards.

## Examples

Typical command usage:

```
device# configure terminal
device(config)# hardware
device(config-hardware)#
```

## History

Release version	Command history
6.0.1a	The description and usage guidelines were modified.

# hardware-profile

Optimizes hardware resources for Keep-Alive Protocol (KAP) profiles, route profiles, or ternary content-addressable memory (TCAM), profiles.

## Syntax

```
hardware-profile kap { custom-profile name | default }
```

```
hardware-profile route-table { default | ipv4-max-arp | ipv4-max-route | ipv4-min-v6 | ipv6-max-nd | ipv6-max-route }
  [ openflow { off | on } ] [ maximum_paths { 8 | 16 | 32 } ]
```

```
hardware-profile tcam { default | dyn-arp-insp | ipv4-acl | ipv4-v6-mcast | ipv4-v6-pbr | ipv4-v6-qos | l2-acl-qos | l2-ipv4-
  acl } [ openflow ]
```

## Command Default

The default hardware profiles are enabled.

## Parameters

### kap

Optimizes hardware resources for KAP profiles, to support hitless failover for the supported protocols.

#### custom-profile *name*

Specifies a custom profile.

#### default

Optimizes basic support for all applications.

### route-table

Optimizes hardware resources for route table profiles.

#### default

Optimizes IPv4 or IPv6 resources for dual-stack operations.

#### ipv4-max-arp

Optimizes resources for the maximum number of IPv4 ARP entries.

#### ipv4-max-route

Optimizes resources for the maximum number of IPv4 routes.

#### ipv4-min-v6

Optimizes resources for IPv4 routes in dual-stack configurations.

#### ipv6-max-nd

Optimizes resources for the maximum number of IPv6 Neighbor Discovery (ND) entries.

#### ipv6-max-route

Optimizes resources for the maximum number of IPv6 routes.

#### openflow

Enables or disables OpenFlow support.

**off**  
Disables OpenFlow.

**on**  
Enables OpenFlow.

**maximum\_paths**  
Specifies 8, 16, or 32 maximum paths.

## tcam

Optimizes hardware resources for TCAM profiles.

**default**  
Optimizes resources with basic support for all applications.

**dyn-arp-insp**  
Optimizes resources for dynamic ARP inspection (DAI).

**ipv4-acl**  
Optimizes resources to allow TCAM usage for IPv4 ACLs.

**ipv4-v6-mcast**  
Optimizes resources for multicast.

**ipv4-v6-pbr**  
Optimizes resources for IPv4 and IPv6 ACLs and policy-based routing tables.

**ipv4-v6-qos**  
Optimizes resources for IPv4 and IPv6 ACLs and QoS.

**l2-acl-qos**  
Optimizes resources for Layer 2 ACLs and QoS.

**l2-ipv4-acl**  
Optimizes resources for Layer 2 IPv4 ACLs.

**openflow**  
Optimizes for OpenFlow support.

## Modes

RBridge ID configuration mode

## Usage Guidelines

### ATTENTION

This is a disruptive command. In order for the last update of the profile configuration to take effect on a switch, the switch must be rebooted by means of the **reload system** command.

The number of potential maximum paths for the Extreme VDX 8770 is 8, 16, or 32; for the Extreme VDX 6740, the number of paths is 8 or 16. The default is 8 for these platforms.

### NOTE

The **acl-max** keyword is supported only on the Extreme VDX 6940-144S. The **ipv4-acl** keyword is supported only on the Extreme VDX 8770 series. If these keywords are configured, a downgrade to a previous version is blocked. For ACL scaling numbers, refer to the chapter "ACLs" in the *Network OS Security Configuration Guide*.

## Examples

The following example optimizes route profiles for IPv4 and IPv6 dual stack operations.

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# hardware-profile route-table default
```

The following example optimizes TCAM resources for multicast.

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# hardware-profile tcam ipv4-v6-mcast
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	This command was modified to add Keep-Alive Protocol (KAP) profiles, OpenFlow, maximum paths, and dynamic ARP (DAI) optimization.
7.0.1	The <b>acl-max</b> and the <b>ipv4-acl</b> keywords were introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

# hardware-profile vlan-classification

Optimizes hardware resources for VLAN classification.

## Syntax

```
hardware-profile vlan-classification { aggregator-basic | aggregator-virtualfabric | aggregator-vxlan-gw | default | tor-virtualfabric | tor-vxlan-gw }
```

## Command Default

The default hardware profiles are enabled.

## Parameters

### aggregator-basic

Optimizes hardware resources for basic support for all applications.

### aggregator-virtualfabric

Optimizes hardware resources for aggregation nodes for Virtual Fabrics.

### aggregator-vxlan-gw

Optimizes hardware resources for aggregation nodes for VXLAN gateways.

### default

Optimizes hardware resources for basic support for all applications.

### tor-virtualfabric

Optimizes hardware resources for Top of Rack (ToR) for Virtual Fabrics.

### tor-vxlan-gw

Optimizes hardware resources for ToR for VXLAN gateways.

## Modes

RBridge ID configuration mode

## Usage Guidelines

### ATTENTION

This is a disruptive command. In order for the last update of the profile configuration to take effect on a switch, the switch has to be rebooted by means of the **reload system** command.

There is no **no** form of this command.

## Examples

To optimize hardware profiles for basic support for all applications:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# hardware-profile vlan-classification default
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.



# hello (LLDP)

Sets the interval between LLDP hello messages.

## Syntax

```
hello seconds
no hello
```

## Command Default

30 seconds

## Parameters

*seconds*  
Valid values range from 4 through 180 seconds.

## Modes

LLDP protocol and profile configuration modes

## Usage Guidelines

The LLDP hello messages can also be configured for a specific LLDP profile. When you apply an LLDP profile on an interface using the **lldp profile** command, it overrides the global configuration. If a profile is not present, then the default global profile is used until you create a valid profile.

Enter **no hello** to return to the default setting.

## Examples

To set the time interval to 10 seconds between the transmissions:

```
device# configure terminal
device (config)# protocol lldp
device(conf-lldp)# hello ?
Possible completions:
<4-180>   Seconds[30 seconds]
device(conf-lldp)# hello 10
```

To set the time interval to 8 seconds between the transmissions for a specific LLDP profile:

```
device(conf-lldp)# profile test1
device(config-profile-test1)# hello 8
device(config-profile-test1)#
```

# hello (UDLD)

Sets the hello transmit interval.

## Syntax

**hello** *hundred\_milliseconds*  
**no hello**

## Command Default

5 is the default value (500 milliseconds).

## Parameters

*hundred\_milliseconds*  
Valid values range from 1 through 60 (in counts of 100 milliseconds).

## Modes

Unidirectional link detection (UDLD) protocol configuration mode

## Usage Guidelines

Use this command to set the time interval between the transmission of hello UDLD PDUs from UDLD-enabled ports.  
Enter **no hello** to return to the default setting.

## Examples

To set the time interval to 2,000 milliseconds between hello UDLD PDU transmissions:

```
device# configure terminal
device(config)# protocol udld
device(config-udld)# hello 20
```

## History

Release version	Command history
16r.1.00	This command was introduced.

## Related Commands

[multiplier \(UDLD\)](#), [protocol udld](#)

# hello-interval

Sets the interval between PDU packets sent from the ELD-enabled edge ports of a cluster.

## Syntax

```
hello-interval interval
```

```
no hello-interval
```

## Command Default

1000 (one second)

## Parameters

*interval*

Number of periods between each PDU. For example, a value of 2000 causes a PDU to be sent every two seconds. Valid values range from 100 through 5000 milliseconds (100 ms through 5 seconds).

## Modes

ELD configuration mode

## Usage Guidelines

Use this command with the **pdu-rx-limit** command to determine the time taken to detect a loop. The time taken to detect a loop is the product of the pdu-rx-limit and the hello interval.

The hello interval must be set to the same value on all clusters that have ELD enabled, otherwise it cannot be predicted which cluster will reach its limit first. The cluster in the loop with the lowest pdu-rx-limit is the cluster where the loop gets broken, assuming that the hello interval is correctly set to the same value on all clusters.

This functionality detects Layer 2 loops only.

Enter **no hello-interval** to return to the default setting.



### CAUTION

Use extreme caution when setting the hello interval value to less than 1 second because it heavily increases the CPU load due to the number of packets transmitted and received. This load depends on the number of ELD instances and other system configuration parameters. Undesirable performance and scalability might occur.

## Examples

To set the PDU interval to 5 seconds:

```
switch(config)# protocol edge-loop-detection
switch(config-eld)# hello-interval 5000
```

hello-interval

To reset the PDU interval to its default value of 1 second:

```
switch(cfg-eld)# no hello-interval 5000
```

## Related Commands

[pdu-rx-limit](#), [protocol edge-loop-detection](#), [show edge-loop-detection globals](#)

# hello-interval (ELD)

This global level configuration defines the interval for sending edge-loop detection (ELD) PDUs.

## Syntax

**hello-interval** *milliseconds*

**no hello-interval** *milliseconds*

## Command Default

The default value is 1000 ms (one second)

## Parameters

*milliseconds*

Interval time in milliseconds. The range is from 100 ms through 5 seconds.

## Modes

ELD configuration mode

## Usage Guidelines

It is the user's responsibility to make sure that the hello interval is set to the same value across the various VCS clouds. Otherwise, the ELD port shutdown will be non-deterministic.

Extreme caution must be taken when setting the hello-interval value to anything less than 1 second, as it will heavily increase the cpu load due to the amount of packets transmitted and received (depending on the number of ELD instances and other system configuration), and might cause undesirable performance and scalability results.

Enter **no hello-interval** *milliseconds* to return to the default setting.

## Examples

To set the PDU hello-interval to 5 seconds:

```
switch(config)# protocol edge-loop-detection
switch(config-eld)# hello-interval 5000
```

To return the PDU hello-interval to the default value (1000 ms):

```
switch(config-eld)# no hello-interval 5000
```

# hello-time

Sets the interval between the hello Bridge Protocol Data Units (BPDUs) sent on an interface.

## Syntax

**hello-time** *seconds*

**no hello-time**

## Command Default

2 seconds

## Parameters

*seconds*

Specifies the time interval between the hello BPDUs sent on an interface. Valid values range from 1 through 10 seconds.

## Modes

Spanning tree configuration mode

## Usage Guidelines

This command configures the spanning-tree bridge hello time, which determines how often the device broadcasts hello messages to other devices.

If the VLAN parameter is not provided, the **hello-time** value is applied globally for all per-VLAN instances. But for the VLANs which have been configured explicitly, the per-VLAN configuration takes precedence over the global configuration. When configuring the **hello-time**, the **max-age** command setting must be greater than the **hello-time** setting. The following relationship should be kept:

$$(2 \times (\text{forward-delay} - 1)) \geq \text{max-age} \geq (2 \times (\text{hello-time} + 1))$$

Enter **no hello-time** to return to the default settings.

The command is the same regardless of which type of STP is enabled.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes. 96

## Examples

To configure spanning tree bridge hello time to 5 seconds:

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# hello-time 5
```

```
device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# hello-time 5
```

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# hello-time 5
```

```
device# configure terminal
device(config)# protocol spanning-tree pvst
device(conf-pvst)# hello-time 5
```

```
device# configure terminal
device(config)# protocol spanning-tree rpvst
device(conf-rpvst)# hello-time 5
```

## Related Commands

[forward-delay](#), [max-age](#)

# hello-timer

Configures the Protocol Independent Multicast (PIM) Hello message periodic interval.

## Syntax

```
hello-timer num  
no hello-timer
```

## Command Default

30 seconds

## Parameters

*num*

The interval value in seconds. Valid values range from 10 through 3600 seconds.

## Modes

RBridge ID configuration mode

PIM router configuration mode

## Usage Guidelines

This command specifies the interval between PIM "Hello" messages.

Enter **no hello-timer** to return to the default settings.

## Examples

To set the Hello interval to 60 seconds.

```
switch(config)# rbridge-id 2  
switch(config-rbridge-id-2)# router pim  
switch(config-pim-router)# hello-timer 60
```

## Related Commands

[router pim](#)



# hold-time

Sets the time that a previously down backup VRRP router, which also must have a higher priority than the current master VRRP router, will wait before assuming mastership of the virtual router.

## Syntax

`hold-time range`

## Command Default

0 seconds

## Parameters

*range*

A value between 1 and 3600 seconds that specifies the time a formerly down backup router waits before assuming mastership of the virtual router.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

The hold-time must be set to a number greater than the default of 0 seconds for this command to take effect.

This command can be used for both VRRP and VRRP-E.

## Examples

To set the hold time to 60 seconds for backup routers in a specific virtual router:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# interface ve 25
device(config-ve-25)# vrrp-extended-group 1
device(config-vrrp-extended-group-1)# hold-time 60
```

## Related Commands

[vrrp-group](#)

# hold-time (Fabric-Virtual-Gateway)

Configures the duration for which the Fabric-Virtual-Gateway session will remain idle before activating the configuration on the system.

## Syntax

**hold-time** *hold-time*

**no hold-time**

## Command Default

None

## Parameters

*hold-time*

The hold time in seconds.

## Modes

Fabric-Virtual-Gateway under VE interface IPv4 or IPv6 configuration mode

## Usage Guidelines

Enter the **no** form of the command to remove the hold-time duration for the IPv4 or IPv6 Fabric-Virtual-Gateway configuration.

## Examples

The following example shows how to configure the hold time.

```
device(config)# interface ve 2000
device(config-ve-2000)# ip fabric-virtual-gateway
device(config-ip-fabric-virtual-gw)# hold-time 30
```

## History

Release version	Command history
5.0.1	This command was introduced.

# host-table aging-mode conversational

Enables conversational address-resolution protocol (ARP) and conversational neighbor discovery (ND). Such enablement improves hardware utilization by programming only active flows into the forwarding plane.

## Syntax

```
host-table aging-mode conversational
no host-table aging-mode conversational
```

## Command Default

Conversational ARP/ND is disabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

You can change the aging-time value from the 300 second default—either before or during enablement—by entering the **host-table aging-time conversational** command.

Conversational ARP/ND can be CPU-intensive.

If conversational ARP/ND is not enabled, make sure that the software ARP/ND cache size is less than the hardware profile limit.

To disable conversational ARP/ND, enter the **no** form of this command.

Upon disablement, the conversational ARP/ND timers no longer apply: All current entries become permanent as do all new entries.

## Examples

The following example enables conversational ARP/ND.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# host-table aging-mode conversational
```

## History

Release version	Command history
7.0.0	This command was introduced.

# host-table aging-time conversational

Specifies a non-default aging-time value for conversational ARP/ND.

## Syntax

`host-table aging-time conversational seconds`

`no host-table aging-time conversational`

## Command Default

If conversational ARP/ND is enabled (by entering the `host-table aging-mode conversational` command), the default aging-time value is 300 seconds.

## Parameters

*seconds*

Specifies the aging-time value for conversational ARP/ND. Values range from 60 through 100000 seconds. The default is 300.

## Modes

RBridge ID configuration mode

## Usage Guidelines

You can modify the aging-time value either before or after enabling conversational ARP/ND.

Note that pre-existing entries age out 30 seconds after enablement, unless they show new activity. The aging-time value applies only to active entries and to new entries.

To restore the default aging-time value of 300 seconds, enter the `no` form of this command.

## Examples

The following example sets the aging-time value to 600 seconds and then enables conversational ARP/ND.

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# host-table aging-time conversational 600
device(config-rbridge-id-2)# host-table aging-mode conversational
```

## History

Release version	Command history
7.0.0	This command was introduced.

# http server

Manages HTTP/HTTPS service on an RBridge.

## Syntax

```
http server { use-vrf vrf-name [ secure-and-plain ] [ shutdown ] }
no http server { use-vrf vrf-name [ secure-and-plain ] [ shutdown ] }
```

## Parameters

**use-vrf** *vrf-name*

Specifies a user-defined VRF.

**secure-and-plain**

Allows the enabling or disabling of both HTTP and HTTPS simultaneously. The HTTPS certificate must be installed for this option to function correctly.

**shutdown**

Disables HTTP/HTTPS service.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **http server use-vrf** *vrf-name* command to associate the HTTP/HTTPS server with the specified VRF and enable service.

Use the **no http server use-vrf** *vrf-name* command to disable HTTP/HTTPS service for the specified VRF and remove its association with that VRF.

Use the **http server use-vrf** *vrf-name* **shutdown** command to disable HTTP or HTTPS service for the specified VRF. When both HTTP and HTTPS are enabled using **secure-and-plain**, then this command disables both HTTP and HTTPS at the same time.

Use the **http server use-vrf** *vrf-name* **secure-and-plain** command to enable both HTTP and HTTPS service for the specified VRF.

Use the **no http server use-vrf** *vrf-name* **secure-and-plain** command to disable HTTP and run HTTPS alone. This command removes **secure-and-plain** option in running-config.

Use the **http server shutdown** to disable HTTP/HTTPS service on management VRF and **no http server shutdown** to enable HTTP/HTTPS service on management VRF.

HTTPS crypto certificates are required to enable HTTPS mode. HTTPS crypto certificates determine whether the service is HTTP or HTTPS. This command is not distributed across the cluster. The RBridge ID of the node is used to configure service on individual nodes.

The use of the **use-vrf** keyword configures HTTP/HTTPS service for the specified VRF only. Service for that VRF is enabled or disabled with no effect on service for other VRFs. The user can disable service for any VRF, including the management VRF.

When **no http server shutdown** is executed and a VRF is not specified by means of the **use-vrf** keyword, the server is enabled or disabled for the management VRF only. Disabling service for the management VRF is allowed, but removing the server's association with the management VRF is not allowed.

The use of the **secure-and-plain** keyword allows you to enable or disable HTTP and HTTPS simultaneously. Without this option, you may only enable HTTP or HTTPS, but not both.

## Examples

The following example creates and enables HTTP/HTTPS service on an RBridge for a user-defined VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# http server use-vrf myvrf
```

The following example creates and enables both HTTP and HTTPS service on an RBridge for a user-defined VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# http server use-vrf myvrf secure-and-plain
```

The following command disables HTTP and run HTTPS alone when both HTTP and HTTPS service are enabled.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# no http server use-vrf myvrf secure-and-plain
```

The following example disables HTTP/HTTPS service (or both HTTP and HTTPS services if both enabled) on an RBridge for a user-defined VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# http server use-vrf myvrf shutdown
```

The following example enables HTTP/HTTPS service on an RBridge for a user-defined VRF when service is disabled.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# no http server use-vrf myvrf shutdown
```

The following example disables HTTP/HTTPS service on an RBridge for a user-defined VRF and removes its association with that VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# no http server use-vrf myvrf
```

The following example disables HTTP/HTTPS service on an RBridge for the management VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# http server shutdown
```

The following example enables HTTP/HTTPS service on an RBridge for the management VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# no http server shutdown
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.
7.1.0	This command was modified to support the <b>secure-and-plain</b> keyword.

# import map evpn

Enables route policy for routes imported from BGP EVPN to a VRF instance.

## Syntax

```
import map { map_name } evpn
no import map { map_name } evpn
```

## Command Default

No route policy is imported.

## Parameters

*map\_name*  
Specifies the name of a route map that is configured by means of the **route-map** command.

## Modes

VRF address-family IPv4 unicast configuration mode  
VRF address-family IPv6 unicast configuration mode

## Usage Guidelines

The import of routes must be configured by means of the **route-target (VRF)** command.

The map is configured per Address Family Identifier (AFI)/Subsequent Address Family Identifier (SAFI) per VRF instance. Once configured, the map is applied to all routes in the VRF for the given AFI/SAFI, and only one map can be applied per VRF at a time. When the second map is applied, the first is overwritten.

Use the **no** form of the command to remove the map.

## Examples

The following example imports a route map in VRF address-family IPv4 unicast configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf red
device(config-vrf-red)# address-family ipv4 unicast
device(vrf-red-ipv4-unicast)# import map map1 evpn
```

The following example removes the map.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf red
device(config-vrf-red)# address-family ipv4 unicast
device(vrf-red-ipv4-unicast)# no import map map1 evpn
```



## History

Release version	Command history
7.1.0	This command was introduced.

# inactivity-timer

Configures the Protocol Independent Multicast (PIM) forwarding-entry inactivity timer.

## Syntax

```
inactivity-timer num  
no inactivity-timer
```

## Command Default

180 seconds

## Parameters

*num*

The entry inactivity timer interval. Valid values range from 60 through 3600 seconds.

## Modes

RBridge ID configuration mode

PIM router configuration mode

## Usage Guidelines

This command specifies the Protocol Independent Multicast (PIM) delay interval until a forwarding entry is considered to be active. At the expiration of this timer, the router deletes a forwarding entry.

Enter **no inactivity-timer** to return to the default setting.

## Examples

To set the inactivity timer to 600 seconds:

```
switch(config)# rbridge-id 2  
switch(config-rbridge-id-2)# router pim  
switch(config-pim-router)# inactivity-timer 600
```

## Related Commands

[router pim](#)

# install-igp-cost

Configures the device to use the IGP cost instead of the default BGP Multi-Exit Discriminator (MED) value as the route cost when the route is added to the Routing Table Manager (RTM).

## Syntax

```
install-igp-cost  
no install-igp-cost
```

## Modes

BGP configuration mode

## Usage Guidelines

By default, BGP uses the BGP MED value as the route cost when the route is added to the RTM. Use this command to change the default to the IGP cost.

The **no** form of the command restores the defaults.

## Examples

The following example configures the device to compare MEDs.

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# router bgp  
device(config-bgp-router)# install-igp-cost
```

# instance

Maps a VLAN to a Multiple Spanning Tree Protocol (MSTP) instance. You can group a set of VLANs to an instance.

## Syntax

```
instance instance_id [ vlan vlan_id | priority priority_id ]
```

```
no instance
```

## Command Default

The priority value is 32768.

## Parameters

*instance\_id*

Specifies the MSTP instance. Valid values range from 1 through 31.

**vlan** *vlan\_id*

Specifies the VLAN to map an MSTP instance. Refer to the Usage Guidelines.

**priority** *priority\_id*

Specifies the priority for the specified instance. Valid values range from 0 through 61440. The priority values can be set only in increments of 4096.

## Modes

Spanning tree MSTP configuration mode

## Usage Guidelines

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

The following rules apply:

- VLANs must be created before mapping to instances.
- The VLAN instance mapping is removed from the configuration if the underlying VLANs are deleted.

Enter **no instance** to remove the VLAN mapping from the MSTP instance.



### CAUTION

This command can be used only after the VLAN is defined.

## Examples

To map a VLAN to an MTSP instance:

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# instance 1 vlan 2,3
device(conf-mstp)# instance 2 vlan 4-6
device(conf-mstp)# instance 1 priority 4096
```

## Related Commands

[show spanning-tree](#)

# interface

Enters the interface configuration mode to configure an interface.

## Syntax

```
interface [ fibrenchannel rbridge-id/slot/port | fcoe vn-number/rbridge-id/front-port-number | <N>gigabitethernet rbridge-id/slot/port | port-channel number | vlan vlan_id ]
```

```
no interface [ port-channel number | vlan vlan_id | fcoe vn-number/rbridge-id/front-port-number ]
```

## Parameters

**fibrenchannel** *rbridge-id/slot/port*

Specifies a valid Fibre Channel port interface.

*rbridge-id*

Specifies the RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**fcoe** *vn-number/rbridge-id/front-port-number*

Specifies a valid FCoE port interface.

*vn-number*

Specifies the VN number for FCoE.

*rbridge-id*

Specifies the routing bridge ID.

*front-port-number*

Specifies the front port number.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies the port-channel number. The number of available channels ranges from 1 through 6144.

**vlan** *vlan\_id*

Specifies the VLAN number. Refer to the Usage Guidelines.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no interface** followed by the appropriate interface identification parameters to disable that interface.

## Examples

To configure a Fibre Channel port on an Extreme VDX 6740 switch:

```
device(config)# interface fibrechannel 66/0/1
device(config-if-fi-66/0/1)#
```

To enter FCoE configuration mode for an interface:

```
device(config)# interface fcoe 1/0/1
device(config-Fcoe-1/0/1)#
```

To enter configuration mode on a 1-Gbps interface on an Extreme VDX device:

```
device(config)# interface gigabitethernet 1/0/1
device(config-if-gi-1/0/1)#
```

# interface (range specification)

Allows a range of values to be entered for some interface configurations.

## Syntax

```
interface { fibrenchannel rbridge-id/slot/port | fcoe vn-number/rbridge-id/front-port-number | <N>gigabitethernet rbridge-id/slot/port | port-channel number | vlan vlan_id | loopback port_number | ve vlan_id }
```

```
no interface { port-channel number | vlan vlan_id | fcoe vn-number/rbridge-id/front-port-number }
```

## Parameters

**fibrenchannel** *rbridge-id/slot/port*

Specifies a valid Fibre Channel port interface.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**fcoe** *vn-number/rbridge-id/front-port-number*

Specifies a valid FCoE port interface.

*vn-number*

Specifies the VN number for FCoE.

*rbridge-id*

Specifies the routing bridge ID.

*front-port-number*

Specifies the front port number.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies the port-channel number. The number of available channels range from 1 through 6144.

**vlan** *vlan\_id*

Specifies the VLAN number. (Refer to the Usage Guidelines.)



**loopback** *port\_number*

Specifies the port number for the loopback interface. The range is 1 through 255.

**ve** *vlan\_id*

Specifies the corresponding VLAN interface that must already be configured before the VE interface can be created. (Refer to the Usage Guidelines.)

## Modes

Global configuration mode (Refer to the Usage Guidelines.)

## Usage Guidelines

Use this command to create or enter the interface configuration mode for an interface or range of interfaces.

Loopback and VE configurations are node specific. The other interfaces that support the use of ranges work the same as shown for VLAN in the examples, except for the following differences:

VE and loopback interfaces also support ranges in RBridge configuration mode.

For example, if you want to create and/or enter VLAN interface configuration mode for VLAN IDs 3 through 8 and VLAN 10 and 12, you would enter the following command in global configuration mode:

```
device(config)# interface vlan 3-8,10,12
```

### NOTE

Do not use a space after a comma or you will receive a syntax error.

You then receive the following prompt:

```
device(config-Vlan-3-8,10,12)#
```

Any command you run from this prompt takes effect on all VLANs that you have specified.

You can use the **no** form of commands on ranges in the same way. For example, if you want to remove the description on VLANs 10 through 15 and VLAN 19 all at the same time, you would enter the following commands in global configuration mode:

```
device(config)# interface vlan 10-15,19
device(config-vlan-10-15,19)# no description
```

### NOTE

The **no** form of the command for deleting interfaces should not be given from the range sub-mode. Exit the range sub-mode before deleting interfaces.

The three gigabit interface types have the following restrictions for range specification in VCS mode:

- Ranges cannot be used for interfaces that belong to multiple slots. However, you can configure a range of interfaces if each interface in the range belongs to the same slot.
- Ranges can be applied only to interfaces that belong to the same RBridge.

Fibrechannel interfaces have the following restrictions for range specification in VCS mode:

- Ranges cannot be used for interfaces that belong to multiple slots. However, you can configure a range of interfaces if each interface in the range belongs to the same slot.
- Ranges can be applied only to interfaces that belong to the same RBridge.

For the fibre channel interface, ranges can be applied only to interfaces that belong to the same RBridge.

An FCoE interface from one node cannot be combined with a bind configuration of a physical port/port-channel that belongs to another node in the cluster. Refer to the Examples.

A set of FCoE ports cannot be bound to the same MAC address.

## Examples

### Examples

To configure binding between a range of FCoE ports (0 to 20 in this example) to the TenGigabitEthernet port 10/0/1:

```
device# configure
device(config)# interface fcoe 1/10/0 - 1/10/20
sw0(conf-Fcoe-1/10/0-20) bind te 10/0/1
```

The FCoE interfaces and the bind configuration of the physical port/port-channel must belong to the same node in the cluster.

An example of an unsupported configuration is:

```
device# configure
device(config)# interface Fcoe 1/10/0 - 1/10/20
device(conf-Fcoe-1/10/0-20)# bind te 20/0/1
```

### Examples

To enter interface subtype configuration mode on a TenGigabitEthernet interface with an RBridge ID of 25 and a slot of 0, with a port range of 1 through 10, 17 through 21, and 24:

```
device(config)# interface tengigabitethernet 25/0/1-10,17-21,24
device(conf-if-te-25/0/1-10,17-21,24)#
```

# interface loopback

Configures a loopback interface.

## Syntax

```
interface loopback port_number  
no interface loopback port_number
```

## Command Default

A loopback interface is not configured.

## Parameters

*port\_number*  
Specifies the port number for the loopback interface. The range is 1 through 255.

## Modes

RBridge ID configuration mode (for VCS)

## Usage Guidelines

A loopback is a logical interface traditionally used to ensure stable routing operations.

The following restrictions apply when the loopback interface is part of an active VXLAN overlay gateway. These restrictions are enforced to maintain consistency across the gateway.

- The loopback interface cannot be deleted.
- The IPv4 address cannot be changed.
- The VRF instance cannot be changed.

You must first use the **no activate** command in VXLAN overlay gateway configuration mode to modify the loopback interfaces. .

Use the no form of this command with a port parameter to remove the specified loopback interface.

## Examples

The following example creates a loopback interface with a port number of 25 for RBridge ID 11. The command is executed in a VCS environment.

```
device(config)# rbridge-id 11  
device(config-rbridge-id-11)# interface loopback 25  
device(config-Loopback-25) #
```

# interface management

Enters configuration mode for the management interface. Also used for binding ACLs to a management interface.

## Syntax

**interface management** *rbridge-id/port*

## Command Default

DHCP is disabled.

IPv6 stateless auto-configuration is disabled.

The speed setting is **auto**.

## Parameters

*rbridge-id/port*

Specifies the management interface to be configured as the *rbridge-id* followed by a slash (/) and the port number. On Top-of-Rack (ToR) switches, the port number for the management port is always 0. On a modular switches with two redundant management modules, you can configure two management ports: 1 and 2.

## Modes

Global configuration mode

## Usage Guidelines

This command supports IP addresses in IPv6 and IPv4 format. This command enters a management interface configuration mode where you can choose configuration parameters for IPv4 and IPv6 addresses.

After you execute this command, the following are among the commands available for management interface configuration:

- **ip address**
- **ip access-group**
- **ip gateway-address**
- **ip icmp**
- **ip route**
- **ipv6 address**
- **ipv6 access-group**
- **ipv6 icmpv6**
- **shutdown**
- **speed**
- **tcp**
- **vrf forwarding**

The **ip gateway-address** command is not available on the Extreme VDX series if the Layer 3 or Advanced Services license is installed. In that case, use the following command sequence:

```
switch(config)# rbridge-id 1
switch(config-rbridge-id-1)# ip route 0000/0 <default_gateway_address>
```

Setting a static IPv4 address and DHCP are mutually exclusive. If DHCP is enabled, you must disable DHCP before you can configure a static IPv4 address.

A static IPv6 address and stateless auto-configuration can coexist.

Auto-configuration is configured chassis-wide and you configure it always under **interface management** *rbridge-id/1*. Once the feature is configured under **interface management** *rbridge-id/1*, it is configured for both management interfaces.

Enter **no ip address** *ipv4\_address/prefix\_len dhcp* to disable DHCP. For other operands, use the **no** form of the command to remove the corresponding configuration.

Enter **no speed** to restore speed parameters to their defaults.

## Examples

The following example configures a management interface with an IPv6 IP address.

```
switch(config)# interface management 1/0
switch(config-Management-1/0)# ipv6 address fd00:60:69bc:832:e61f:13ff:fe67:4b94/64
```

The following example sets the interface to 100-Mbps Full Duplex.

```
switch(config-Management-1/0)# speed 100
```

The following example applies ACLs to management interfaces.

```
switch(config)# interface Management 1/1
switch(config-Management-1/1)# ip access-group stdACL3 in
switch(config-Management-1/1)# ipv6 access-group stdV6ACL1 in
switch(config-Management-1/1)# exit
switch(config)# interface Management 1/2
switch(config-Management-1/2)# ip access-group extdACL5 in
switch(config-Management-1/2)# exit
```

The following example enables DHCP for IPv4 addresses.

```
switch(config)# interface Management 1/1
switch(config-Management-1/1)# ip address dhcp
```

The following example enables DHCP for IPv6 addresses.

```
switch(config)# interface Management 1/1
switch(config-Management-1/1)# ipv6 address dhcp
```

The following example applies an ACL to management interface 1/1.

```
switch(config)# interface management
switch(config)# interface management 1/1
Entering configuration mode terminal
switch(config-Management-1/1)# ip access-group stdACL1 in
```

# interface ve

Configures a virtual Ethernet (VE) interface.

## Syntax

```
interface ve vlan_id
no interface ve vlan_id
```

## Parameters

*vlan\_id*

Specifies the corresponding VLAN interface that must already be configured before the VE interface can be created. Refer to the Usage Guidelines.

## Modes

RBridge ID configuration mode (for VCS)

Global configuration mode

## Usage Guidelines

Before you can configure a VE interface, you must configure a VLAN interface. The corresponding VE interface must use the same VLAN ID you used to configure the VLAN.

Use the **no** form of the command to remove the VE interface.



### CAUTION

If no RBridge ID is configured on the switch, deleting the VE interface will cause a spike in CPU usage. To prevent this, configure an RBridge ID before deleting the VE interface.

## Examples

The following example shows the steps needed to create a VE interface with the VLAN ID of 56 for RBridge ID 11. This example is for a VCS environment, and assumes that the VLAN 56 interface has already been created.

```
device(config)# rbridge-id 11
device(config-rbridge-id-11)# interface ve 56
device(config-Ve-56)#
```

The following example shows the steps needed to create a VE interface with the VLAN ID of 4093.

```
device# configure
device(config)# interface ve 4093
device(config-Ve-4093)#
```

## Related Commands

[interface](#), [interface vlan](#)

# interface vlan

Allows the user to create 802.1Q VLANs, as well as service or transport VFs in a Virtual Fabrics context.

## Syntax

```
interface vlan vlan_id
```

```
no interface vlan vlan_id
```

## Command Default

VLAN 1 is predefined on the switch.

## Parameters

*vlan\_id*

Specifies the VLAN interface to configure. The range is from 1 through 8191. (Refer to the Usage Guidelines.)

## Modes

Global configuration mode

## Usage Guidelines

Use this command to configure a VLAN interface. This command applies to both 802.1Q VLANs (whose VLAN IDs range from 1 through 4095) and service or transport VFs (whose VLAN IDs range from 4096 through 8191). VLAN IDs 3964 through 4090 are internally-reserved VLAN IDs. However, the **reserved-vlan** command can modify this range. By default all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1).

### NOTE

For the Extreme VDX 67XX switches, the default reserved space is 128 VLANs, and is equal to sum of the number of maximum allowed port channels and the number of interfaces. Presently, VLANs from 3960 through 4090 are reserved.

### NOTE

For the Extreme VDX 8770 switches, the default reserved VLAN space is 4. VLANs 4087 through 4090 are reserved.

To support multi-tenancy, assigning VLAN IDs from 4096 through 8191 creates service or transport VFs that are unique within a local VCS Fabric but that cannot extend to another VCS Fabric.

All of the ports on the switch are a part of the default VLAN 1.

Make sure your converged mode interface is not configured to classify untagged packets to the same VLAN as the incoming VLAN-tagged packets. By configuring a converged interface to classify untagged packets (by using classifiers or the default port *vlan\_id*) to the same VLAN as VLAN-tagged packets coming into the interface, the FCoE hardware sends out untagged packets to the CNA. These packets may be dropped, disrupting communications.

For service or transport VFs to be implemented in a Virtual Fabrics context, the user must execute the **vcs virtual-fabric enable** command in global configuration mode.

Enter **no interface vlan *vlan\_id*** to delete a VLAN interface. This will also delete the corresponding virtual Ethernet (VE) interface.

## Examples

To create a VLAN with an ID of 56:

```
switch(config)# interface vlan 56
switch(config-vlan-56)#
```

To create a classified VLAN (with an ID from 4096 through 8191):

```
switch(config)# interface vlan 5000
switch(config-vlan-5000)#
```



# interval

For an implementation of an event-handler profile, specifies the number of seconds between iterations of an event-handler action, if triggered.

## Syntax

`interval seconds`

`no interval`

## Command Default

Iterations occur with no interval between them.

## Parameters

*seconds*

Specifies the number of seconds between iterations of an event-handler action, if triggered. Valid values are 0 or a positive integer.

## Modes

Event-handler activation mode

## Usage Guidelines

The **interval** command is effective only if the **iterations** value is non-zero.

The **no** form of this command resets the **interval** setting to the default 0 seconds.

## Examples

The following example sets the number of iterations to 3 and specifies an interval of 10 seconds between each iteration.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# iterations 3
device(config-activate-eventHandler1)# interval 10
```

The following example resets **interval** to the default value of 0 seconds.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# no interval
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ip access-group

Applies rules specified in an IPv4 access control list (ACL) to traffic entering or exiting an interface.

## Syntax

```
ip access-group ACLname { in | out } [ switched | routed ]
no ip access-group ACLname { in | out } [ switched | routed ]
```

## Parameters

*ACLname*

Specifies the name of the standard or extended IPv4 access list.

**in | out**

Specifies the binding direction (ingress or egress).

**switched**

Filter only switched traffic. This parameter is not valid for management or overlay-gateway interfaces.

**routed**

Filter only routed traffic. This parameter is not valid for management or overlay-gateway interfaces.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to apply an IPv4 ACL to one of the following interface types:

- User interfaces
  - Physical Ethernet interfaces
  - Logical interfaces (LAGs)
  - Virtual Ethernet interfaces (VEs)
- All supported management interfaces
- Overlay gateways

You can apply a maximum of six ACLs to a user interface, as follows:

- One ingress MAC ACL—if the interface is in switchport or overlay-gateway mode
- One egress MAC ACL—if the interface is in switchport mode
- One ingress IPv4 ACL
- One egress IPv4 ACL
- One ingress IPv6 ACL
- One egress IPv6 ACL

You can apply a maximum of two ACLs to a management interface, as follows:

- One ingress IPv4 ACL

- One ingress IPv6 ACL

You can apply a maximum of three ACLs to an overlay gateway, as follows:

- One ingress MAC ACL
- One ingress IPv4 ACL
- One ingress IPv6 ACL

You can apply an ACL to multiple interfaces. And you can apply an ACL twice—ingress and egress—to a given user interface.

If you do not specify **switched** or **routed**, the ACL applies both to switched and routed traffic.

To remove an ACL from an interface, enter the **no** form of this command.

## Examples

The following example applies an ingress IP ACL on a 10-gigabit Ethernet interface.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# ip access-group ipacl2 in
```

The following example removes an ingress IP ACL from a 10-gigabit Ethernet interface:

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# no ip access-group ipacl2 in
```

# ip access-list

Creates a standard or extended IPv4 access control list (ACL). In ACLs, you can define rules that permit or deny network traffic based on criteria that you specify.

## Syntax

```
ip access-list { standard | extended } ACLname
no ip access-list { standard | extended } ACLname
```

## Parameters

**standard | extended**

Specifies one of the following types of access lists:

**standard**

Contains rules that permit or deny traffic based on source addresses that you specify. The rules are applicable to all ports of the specified addresses.

**extended**

Contains rules that permit or deny traffic according to source and destination addresses, as well as other parameters. For example, you can also filter by port, protocol (TCP or UDP), and TCP flags.

*ACLname*

Specifies an ACL name unique among all ACLs (Layer 2 and Layer 3). The name can be up to 63 characters in length, and must begin with an alphanumeric character. No special characters are allowed, except for the underscore and hyphen.

## Modes

Global configuration mode

## Usage Guidelines

An ACL name can be up to 63 characters long, and must begin with a-z, A-Z or 0-9. You can also use underscore (\_) or hyphen (-) in an ACL name, but not as the first character.

After you create an ACL, use the **seq** command to create filtering rules for that ACL.

An ACL starts functioning only after it is applied to an interface, using the **access-group** command.

To delete an ACL, use the **no access-list** command. You can delete an ACL only after you first remove it from all interfaces to which it is applied, using the **no access-group** command.

## Examples

The following example creates an IPv4 standard ACL.

```
device# configure
device(config)# ip access-list standard stdACL3
```

The following example creates an IPv4 extended ACL.

```
device# configure
device(config)# ip access-list extended extdACL5
```

The following example creates rules on an IPv4 standard ACL.

```
device# configure
device(config)# ip access-list standard stdACL3
device(config-ipacl-std)# seq 5 permit host 10.20.33.4
device(config-ipacl-std)# seq 15 deny any
```

The following example deletes an IPv4 ACL.

```
device# configure
device(config)# no ip access-list standard stdACL3
```

# ip address

Configures an IP address on an interface.

## Syntax

```
ip address ip-address/mask [ ] [ secondary ] [ ospf-ignore ] ospf-active
no ip address
```

## Parameters

*ip-address*

Specifies the IP address.

*mask*

Specifies the mask for the associated IP subnet. Valid values are integers from 1 through 31. Dotted-decimal is not supported.

**secondary**

Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.

**ospf-ignore**

Disables adjacency formation with OSPF neighbors and disables advertisement of the interface to OSPF.

**ospf-passive**

Disables adjacency formation with OSPF neighbors but does not disable advertisement of the interface to OSPF.

## Modes

Interface configuration mode

Management interface configuration mode

## Usage Guidelines

- Use this command to configure a primary or secondary IP address for a specific interface. You can also use this command to prevent OSPF from running on specified subnets. Multiple primary IP addresses are supported on an interface.
- You can use this command to configure a primary or secondary IP address for a management interface.
- For a management interface, only one primary IP address is supported. Secondary IP addresses are not supported.
- A primary IP address cannot overlap with a previously configured IP subnet.
- A primary IP address must be configured before you configure a secondary IP address in the same subnet.
- To remove the configured static or DHCP address, enter **no ip address**. This resets the address to 0.0.0.0/0.
- The **no** form of the command removes a specific IP address from the interface.

## Examples

The following example configures a primary IP address on a specified Ethernet interface.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# ip address 10.1.1.1/24
```

The following example replaces the primary IP address of an interface. .

The following example configures a secondary IP address on a specified Ethernet interface.

```
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# ip address 10.1.1.2/28 secondary
```

## Related Commands

[interface \(range specification\)](#), [interface ve](#)



# ip address (NSX controller configuration)

Configures the IP address, port and connection method settings for an NSX controller connection profile.

## Syntax

```
ip address ip-address [ method { ssl | tcp } ] [ port port_number ]
```

## Parameters

**ip address** *ip-address*

IP address of the NSX Controller cluster. Only IPv4 addresses are allowed. This address is used to open a connection to the NSX Controller for Open vSwitch Database Management Protocol (OVSDB) exchange.

**method**

Specifies the connection method for this profile.

**ssl**

Specifies that a Secure Sockets Layer connection will be used. This is the default connection method.

**tcp**

Specifies that a transmission control protocol will be used.

**port** *port\_number*

Specifies the port number for the NSX controller. The range is 1-65535. The default 6632.

## Modes

NSX controller configuration mode

## Usage Guidelines

The VXLAN gateway must be in shutdown state.

## Examples

The following example shows how to enter NSX controller configuration mode for the already created NSX controller connection profile called profile1, then how to create the IP address, set the method to TCP, and designate the port of 25.

```
device# configure terminal
device(config)# nsx-controller profile1
device(config-nsx-controller-profile1)# ip address 10.21.83.188 method tcp port 25
```

# ip address (OpenFlow)

Specifies an IPv4 address for an OpenFlow controller, as well as a port number through which to connect and a connection method.

## Syntax

```
ip address { ip-address } [ method { no-ssl | ssl } ] [ port port-number ] [ use-vrf vrf-name ]
no ip address { ip-address } [ method { no-ssl | ssl } ] [ port port-number ] [ use-vrf vrf-name ]
```

## Command Default

See the Parameters and Usage Guidelines.

## Parameters

*ip-address*

IPv4 address of the OpenFlow controller in dotted-decimal format.

**method**

Specifies the method by which to connect to the OpenFlow controller, SSL or TCP. Currently only SSL is supported..

**no-ssl**

Specifies a TCP connection.

**ssl**

Specifies an SSL connection (the default).

**port** *port-number*

Specifies the port through which to connect. See the Usage Guidelines.

**use-vrf** *vrf-name*

Specifies a VRF through which to connect to the controller. See the Usage Guidelines.

## Modes

OpenFlow controller configuration mode

## Usage Guidelines

Currently, only SSL is supported for the connection method. The TCP port is usually limited to either 6633 or 6653 and must match the port configured on the controller. The default port is 6633.

Use the **no** form of this command to delete the IP address and its associated configurations. You cannot remove the IP address and associated configurations from an active controller.

The options available under this command are also available under the **openflow-controller** command.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

This example specifies an IPv4 address, a TCP connection, the nondefault port, and the VRF "myvrf".

```
device(config)# openflow-controller mycontroller
device(config-openflow-controller-mycontroller)# ip address 10.24.82.10 method no-ssl
port 6653 use-vrf myvrf
device(config-openflow-controller-mycontroller)#
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ip address (VXLAN)

Specifies the destination IPv4 address of a tunnel in VXLAN overlay gateway configurations.

## Syntax

**ip address** *IPv4\_address*

**no ip address** [*IPv4\_address*]

## Parameters

*IPv4\_address*

IPv4 address of the destination tunnel.

## Modes

VXLAN overlay gateway site configuration mode

## Usage Guidelines

This command creates a tunnel when the parent overlay gateway is attached to one or more RBridges. The tunnel mode and the source IP address are derived from the parent overlay gateway.

To change an IP addresses, you must first remove the existing address, by means of the **no ip address** *IPv4\_address* or the **no ip address** commands. This also deletes all tunnels to the site.

Only one IPv4 address is allowed. The following IPv4 addresses are not allowed:

- Broadcast addresses (0.0.0.0 through 0.255.255.255)
- Localhost loopback addresses (127.0.0.0 through 127.255.255.255)
- Multicast addresses (224.0.0.0 through 239.255.255.255)
- Reserved addresses (240.0.0.0 through 255.255.,255.255)

## Examples

To specify an IPv4 address of a destination tunnel:

```
switch(config)# overlay-gateway gateway1
switch(config-overlay-gw-gateway1)# site mysite
switch(config-overlay-gw-gateway1-site-mysite)# ip address 10.11.12.13
```

## Related Commands

[overlay-gateway](#), [site](#)

# ip anycast-address

Configures the IPv4 anycast address on an interface.

## Syntax

**ip anycast-address** *ip-address / mask*

**no ip anycast-address** *ip-address / mask*

## Command Default

No IPv4 anycast address is defined.

## Parameters

*ip-address / mask*

Specifies the IPv4 anycast address and mask. A mask value is required.

## Modes

Virtual ethernet (VE) configuration mode

## Usage Guidelines

This command is supported only if ARP suppression is enabled on the VE.

To delete a configured IPv4 anycast address, use the **no** form of this command.

## Examples

The following example specifies an IPv4 anycast address on VE 10.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# interface ve 10
device(config-rbridge-ve-10)# ip anycast-address 2.2.2.2/24
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ip anycast-gateway-mac

Configures the IPv4 anycast-gateway MAC address.

## Syntax

```
ip anycast-gateway-mac { default-mac | mac-address }
no ip anycast-gateway-mac { default-mac | mac-address }
```

## Command Default

No IPv4 anycast-gateway MAC address is defined.

## Parameters

### default-mac

Sets the IPv4 anycast-gateway MAC address to 02e0.5200.0100.

### mac-address

Specifies a non-default IPv4 anycast-gateway MAC address.

## Modes

RBridge-ID configuration mode

## Usage Guidelines

The first three bytes (six digits) of a non-default IPv4 anycast-gateway MAC address must be identical with the first three bytes of a corresponding IPv6 anycast-gateway MAC address. For example if the IPv4 anycast-gateway MAC address is 2222.2244.4444, the IPv6 address could be 2222.2266.6666.

To change a configured MAC address, first delete the current address.

When this feature is enabled in an IP Fabric, to prevent ARP/ND broadcast across the network, the user should enable ARP/ND suppression on the respective VLANs.

To delete the configured IPv4 MAC address, use the **no** form of this command.

## Examples

The following example specifies a default IPv4 anycast-gateway MAC address.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ip anycast-gateway-mac default-mac
```

The following example specifies a non-default IPv4 anycast-gateway MAC address.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ip anycast-gateway-mac 2222.2244.4444
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ip arp-aging-timeout

Sets how long a dynamic Address Resolution Protocol (ARP) entry stays in the ARP cache. The aging timer is reset each time an ARP reply is received.

## Syntax

```
ip arp-aging-timeout value
```

```
no ip arp-aging-timeout
```

## Command Default

ARP aging timeout is globally enabled and set to 240 minutes.

## Parameters

*value*

Specifies how long an ARP entry stays in the ARP cache. Values range from 0 through 240 minutes.

## Modes

Interface subtype configuration mode

## Usage Guidelines

When the device places an entry in the ARP cache, the device also starts an aging timer for the entry. The aging timer ensures that the ARP cache does not retain learned entries that are no longer valid. An entry can become invalid when the device with the MAC address of the entry is no longer on the network.

The aging timer is reset each time an ARP reply is received.

Aging out affects dynamic (learned) entries only. Static entries do not age out.

You can modify the ARP aging timeout only at the interface level, but not at the global level.

To prevent entries from aging out, enter **ip arp-aging-timeout 0**.

The **no** form of the command restores the default aging timeout of 240 minutes.

## Examples

The following command sets the ARP aging timeout to 100 minutes on an interface.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# ip arp-aging-timeout 100
```

The following command restores the ARP aging timeout to the default value on an interface.

```
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# no ip arp-aging-timeout
```



# ip arp inspection

Enables dynamic ARP inspection (DAI) on a VLAN.

## Syntax

```
ip arp inspection
no ip arp inspection
```

## Command Default

The DAI feature is disabled.

## Modes

VLAN configuration mode

## Usage Guidelines

To disable ARP inspection, use the **no** form of this command.

On untrusted interfaces of DAI-enabled VLANs, incoming ARP packets from permitted IP/MAC addresses are accepted only if all of the following steps were performed:

- Create the ACL, using the **arp access-list** command.
- In the ACL, create one or more rules, using the **permit ip host** command. Each rule specifies an IP/MAC address-pair.
- Apply the ACL to one or more VLANs, using the **ip arp inspection filter** command.
- Enable DAI on such VLANs, using the **ip arp inspection** command.

## Examples

The following example applies ARP\_ACL\_01 to VLAN 200 and enables DAI.

```
device# configure terminal
device(conf)# interface vlan 200
device(conf-if-vlan-200)# ip arp inspection filter ARP_ACL_01
device(conf-if-vlan-200)# ip arp inspection
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ip arp inspection filter

Applies an address resolution protocol (ARP) access list (ACL) to a VLAN, which is one of the steps implementing dynamic ARP inspection (DAI) on a VLAN.

## Syntax

```
ip arp inspection filter ACL_name
```

```
no ip arp inspection filter
```

## Command Default

No ARP-ACL is applied.

## Parameters

*ACL\_name*

Specifies which ACL is applied to the VLAN.

## Modes

VLAN configuration mode

## Usage Guidelines

To remove the current ARP-ACL from the VLAN, use the **no** form of this command.

On untrusted interfaces of DAI-enabled VLANs, incoming ARP packets from permitted IP/MAC addresses are accepted only if all of the following steps were performed:

- Create the ACL, using the **arp access-list** command.
- In the ACL, create one or more rules, using the **permit ip host** command. Each rule specifies an IP/MAC address-pair.
- Apply the ACL to one or more VLANs, using the **ip arp inspection filter** command.
- Enable DAI on such VLANs, using the **ip arp inspection** command.

## Examples

The following example applies an ARP-ACL named ARP\_ACL\_01 to VLAN 200.

```
device# configure terminal
device(conf)# interface vlan 200
device(conf-if-vlan-200)# ip arp inspection filter ARP_ACL_01
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ip arp inspection logging acl-match

Specifies whether or not to enable dynamic ARP inspection (DAI) logging.

## Syntax

```
ip arp inspection logging acl-match [ matchlog | none ]
```

## Command Default

DAI logging is disabled.

## Parameters

### logging

Enables or disables DAI logging of permitted packets on the current VLAN.

### matchlog

Required for enablement of DAI logging. For additional requirements, refer to the Usage Guidelines.

### none

Disables DAI logging.

## Modes

VLAN configuration mode

## Usage Guidelines

To disable DAI logging, use the **ip arp inspection logging acl-match none** command.

The following conditions are required to enable DAI logging:

- In VLAN configuration mode, enter the **ip arp inspection logging acl-match matchlog** command.
- Apply to the VLAN an ARP-ACL with at least one **permit** statement containing the **log** parameter.
- Enter the **terminal monitor** command.

## Examples

The following example applies ARP\_ACL\_01 to VLAN 200, enables DAI logging on VLAN 200, enables DAI, and displays the log.

```
device# configure terminal
device(conf)# interface vlan 200
device(conf-if-vlan-200)# ip arp inspection filter ARP_ACL_01
device(conf-if-vlan-200)# ip arp inspection acl-match matchlog
device(conf-if-vlan-200)# ip arp inspection
device(conf-if-vlan-200)# end
device# terminal monitor
Terminal monitoring is enabled.
device# 015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
2015/03/11-18:47:06 PERMIT: arp Packet with srcIp=1.1.1.1, srcMac=0001.0001.0001,
dstIp=10.1.1.1,dstMac=ffff.ffff.ffff on Vlan: 11
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ip arp inspection trust

Configures the interface as trusted, for all VLANs configured on it, which is one of the steps implementing dynamic ARP inspection (DAI) on a VLAN or VE.

## Syntax

```
ip arp inspection trust
no ip arp inspection trust
```

## Command Default

The interface is untrusted.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command is supported only on Layer 2 physical or port-channel interfaces.

On trusted interfaces, all incoming ARP packets are accepted.

On untrusted interfaces of DAI-enabled VLANs, incoming ARP packets from permitted IP/MAC addresses are accepted only if all of the following steps were performed:

- Create the ACL, using the **arp access-list** command.
- In the ACL, create one or more rules, using the **permit ip host** command. Each rule specifies an IP/MAC address-pair.
- Apply the ACL to one or more VLANs, using the **ip arp inspection filter** command.
- Enable DAI on such VLANs, using the **ip arp inspection** command.

To configure the interface as untrusted, use the **no** form of this command.

## Examples

The following example configures a ten gigabit Ethernet interface as trusted.

```
device# configure terminal
device(conf)# interface tengigabitethernet 2/1/1
device(conf-if-te-2/1/1)# ip arp inspection trust
```

The following example configures a port-channel interface as untrusted.

```
device# configure terminal
device(conf)# interface port-channel 171
device(config-Port-channel-171)# no ip arp inspection trust
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ip arp learn-any

Enables address-resolution protocol (ARP) learning from any ARP request.

## Syntax

```
ip arp learn-any
no ip arp learn-any
```

## Command Default

Default ARP learning.

## Modes

VE configuration mode

## Usage Guidelines

This command is effective only on a router port.

This command enables learning from any ARP request (not necessarily targeted to my ip address).

To reset default ARP learning, use the **no** form of this command.

## Examples

The following example enables learn-any on VE 100.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# interface ve 100
device(config-rbridge-Ve-100)# ip arp learn-any
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ip as-path access-list

Configures an AS-path access control list (ACL), specifies the community name, and whether to permit or deny traffic.

## Syntax

```
ip as-path access-list string [ seq seq-value ] [ deny regular-expression | permit regular-expression ]  
no ip as-path access-list string [ seq seq-value ] [ deny regular-expression | permit regular-expression ]
```

## Command Default

This option is disabled.

## Parameters

*string*

ACL name.

*seq-value*

Sequence number as defined by the **seq** command.

*regular-expression*

A string inside quotes.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This command accepts a regular expression that must be enclosed in quotes.

Use the **no** form of this command to restore the default.

## Examples

To create an AS-path ACL:

```
switch(config)# rbridge-id 10  
switch(config-rbridge-id-10)# ip as-path access-list seq 10 permit "myaspath"
```

## Related Commands

[seq \(IPv6 extended ACLs\)](#), [seq \(IPv6 standard ACLs\)](#)



# ip community-list extended

Configures a community access control list (ACL), specifies the community name, and whether to permit or deny traffic, including through the use of a regular expression.

## Syntax

```
ip community-list extended community-list-name { deny string | permit string } [ seq seq ] [ internet | local-as | no-advertise | no-export ]
```

```
no ip community-list extended community-list-name
```

## Command Default

This option is disabled.

## Parameters

*community-list-name*

Range is from 1 through 32 ASCII characters.

*string*

An ordered community-list regular expression.

*seq*

Sequence number. Range is from 1 through 65535.

**internet**

The Internet community.

**no-export**

Community of sub-ASs within a confederation. Routes with this community can be exported to other sub-ASs in the same confederation but not outside the confederation to other ASs or otherwise sent to EBGP neighbors.

**local-as**

Local sub-AS within the confederation. Routes with this community can be advertised only within the local sub-AS.

**no-advertise**

Routes with this community cannot be advertised to any other BGP4 devices at all.

*regular-expression*

A string enclosed in quotes.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Unlike a standard community list, this command does accept a regular expression as long as the string is enclosed in quotes.

Use the **no** form of this command to restore the default.

## Examples

To create an extended community list:

```
switch(config)# rbridge-id 10  
switch(config-rbridge-id-10)# ip community-list extended seq 10 permit "mycommunity"
```

## Related Commands

[ip community-list standard, seq \(IPv6 extended ACLs\)](#)

# ip community-list standard

Configures a community access control list (ACL), specifies the community number or type, and whether to permit or deny traffic.

## Syntax

```
ip community-list standard community-list-name { deny [ community-number | AA:NN ] | permit community-number } [ seq
  seq-value ] [ internet | local-as | no-advertise | no-export ]
```

```
no ip community-list standard community-list-name
```

## Command Default

This option is disabled.

## Parameters

*community-list-name*

Range is from 1 through 32 ASCII characters.

*community-number*

A community number. Range is from 1 through 4294967295.

*AA : NN*

Autonomous system number and network number, configured as 2-byte numbers separated by a colon.

*seq*

Sequence number. Range is from 1 through 65535.

**internet**

The Internet community.

**no-export**

Community of sub-ASs within a confederation. Routes with this community can be exported to other sub-ASs in the same confederation but not outside the confederation to other ASs or otherwise sent to EBGp neighbors.

**local-as**

Local sub-AS within the confederation. Routes with this community can be advertised only within the local sub-AS.

**no-advertise**

Routes with this community cannot be advertised to any other BGP4 devices at all.

## Modes

RBridge ID configuration mode

## Usage Guidelines

A standard community list does not accept a regular expression.

Use the **no** form of this command to restore the default.

## Examples

To create a standard community list:

```
switch(config)# rbridge-id 10
switch(config-rbridge-id-10)# ip community-list standard seq 10 permit local-as
```

## Related Commands

[ip community-list extended](#), [seq \(IPv6 standard ACLs\)](#)

# ip dhcp relay address

Configures the IP DHCP Relay on a Layer 3 interface.

## Syntax

```
ip dhcp relay address ip-addr [ use-vrf vrf-name ]
```

## Parameters

*ip-addr*

IPv4 address of the DHCP server where the DHCP client requests are to be forwarded.

**use-vrf**

Use this option if the VRF where the DHCP server is located is different from the VRF of the interface where the client is connected.

*vrf-name*

VRF name.

## Modes

Interface configuration mode

## Usage Guidelines

This command uses the IPv4 address of the DHCP server where the DHCP client requests are to be forwarded.

You can configure the address on a virtual Ethernet (VE) or a physical GigabitEthernet, TenGigabitEthernet, or FortyGigabitEthernet interface

Enter the command while in interface configuration mode for a VE or physical interface where you want to configure the IP DHCP Relay. Configure up to four DHCP server IP addresses per interface.

Use the **no** version of this command to remove the IP DHCP relay from the interface. If the **use-vrf** option is not used, it is assumed that the DHCP server and interface where the client is connected are on the same VRF.

## Examples

To configure an IP DHCP Relay address on a VE interface:

```
device(config)# rbridge-id 1
device(config-rbridge-id-1)interface ve 101
device(config-rbridge-id-Ve-101)# ip dhcp relay address 100.1.1.2
device(config-rbridge-id-Ve-101)# ip dhcp relay address 12.3.4.6
```

To configure an IP DHCP Relay address on an interface if the DHCP server is on a different VRF than the interface where the client connects:

```
device# config
Entering configuration mode terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# interface ve 103
device(config-rbridge-id-Ve-103)# ip dhcp relay address 3.1.2.255 use-vrf blue
```

ip dhcp relay address

## Related Commands

[show ip dhcp relay address interface](#)

# ip dhcp relay gateway address

Configures the IP DHCP Relay on a Layer 3 gateway interface.

## Syntax

```
ip dhcp relay gateway address ip-addr
no ip dhcp relay gateway address ip-addr
```

## Parameters

*ip-addr*

IPv4 gateway address of the DHCP server where the DHCP client requests are to be forwarded.

## Modes

Interface configuration mode

## Usage Guidelines

Use this command to configure the IP DHCP Relay on the switch Layer 3 gateway interface using the IPv4 address of the DHCP server where the DHCP client requests are to be forwarded.

You can configure the gateway address on a virtual Ethernet (VE) or a physical GigabitEthernet, TenGigabitEthernet, or FortyGigabitEthernet interface. Enter the command while in interface configuration mode for a VE or physical interface where you want to configure the IP DHCP Relay. Configure up to four DHCP server IP addresses per interface.

Use the **no** version of this command to remove the IP DHCP Relay from the interface.

## Examples

To configure an IP DHCP Relay gateway address on a TenGigabitEthernet interface:

```
sw0(config)# interface tengigabitethernet 2/3/1
sw0(conf-if-te-2/3/1)# ip dhcp relay gateway address 100.1.1.2
```

To configure an IP DHCP Relay address on a global VE interface:

```
sw0(config)# interface ve 100
sw0(config-Ve-100)# ip dhcp relay gateway address 100.1.1.2
```

To configure an IP DHCP Relay address on an RBridge VE interface:

```
switch(config)# rbridge-id 1
switch(config-rbridge-id-1)interface ve 101
switch(config-rbridge-id-Ve-101)# ip dhcp relay gateway address 100.1.1.2
switch(config-rbridge-id-Ve-101)# ip dhcp relay gateway address 12.3.4.6
```

## History

Release version	Command history
4.1.3	This command was introduced.
6.0.1a	This command was modified to remove reference to "standalone" mode.



# ip dhcp relay information option

Toggles the DHCP Relay Information Option (option-82) present in the DHCP client and server packets.

## Syntax

```
ip dhcp relay information option
no ip dhcp relay information option
```

## Command Default

DHCP relay information option is disabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no ip dhcp relay information option** command to deactivate this option.

DHCP broadcast requests are relayed by the DHCP Relay Agent to the DHCP Server. The DHCP replies are unicasted to the DHCP Relay Agent, which relays them back to the DHCP client.

DHCP Relay Agent inserts the "giaddr" field in the relayed DHCP packets, so that the DHCP server may identify the pool to be used for the request. The choice of the pool is made based on the "giaddr" field, or the incoming interface, if the "giaddr" is missing or zero.

When enabled, option-82 information is inserted by the Relay Agent before relaying the DHCP client packets to the Server. This information allows the DHCP server to select an IP address or other parameter. The DHCP Server echoes the option-82 in the reply packets. DHCP Relay Agent validates and removes the option-82 and sends the response to the DHCP Client.

Activating the relay information option to the DHCP client requests helps address the following security aspects:

- Identifies which circuit replies should be forwarded.
- Prevents DHCP IP exhaustion attacks.
- Permanently assigns an IP address to a particular user or modem.
- Prevents the spoofing of client identifier fields that are used to assign IP addresses.
- Prevents denial of service by preventing the spoofing other client's MAC addresses.

Use the description command to configure the string to be transmitted by option-82, under the intended interface configured with DHCP Relay configuration:

- For a VE interface: `device(config-vlan-100)# description RequiredString`
- For a Physical Interface: `device(conf-if-gi-1/0/12)# description RequiredString`

The DHCP Server can be configured to parse for the following cases:

- Match on only String
- Match on String and VLAN
- Match on String, VLAN and MAC address

## Examples

The following example enables the relay information option.

```

device(config)# rbridge-id-1
device(config-rbridge-id-1)# ip dhcp relay information option
DHCP Relay Agent Information Option is enabled.
device(config-rbridge-id-1)# do show ip dhcp relay address
DHCP Relay Agent Information Option Enabled
                                Rbridge Id:    2
                                -----
Interface                        Relay Address          VRF Name
-----
Te 2/2/1                        10.1.1.1              Blue
Te 2/4/2                        20.1.1.1              Blue
Te 2/5/4                        30.1.1.1              Default-vrf
Ve 100                          40.1.1.1              Green

```

## History

Release version	Command history
6.0.2a	This command was introduced.

## Related Commands

[description \(interfaces\)](#), [show ip dhcp relay address interface](#)

# ip directed-broadcast

Enables directed broadcasts on an interface. A directed broadcast is an IP broadcast to all devices within a directly attached network or subnet.

## Syntax

```
ip directed-broadcast
no ip directed-broadcast
```

## Command Default

Directed broadcast is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

To disable directed broadcasts on an interface, enter the **no** form of this command.

## Examples

The following example enables directed broadcasts on an Ethernet interface.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# ip directed-broadcast
```

The following example disables directed broadcasts on an Ethernet interface.

```
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# no ip directed-broadcast
```

## Related Commands

[interface](#), [interface ve](#)

# ip dns

Configures the Domain Name System (DNS) domain name and the primary and secondary name server IP addresses.

## Syntax

```
ip dns { domain-name domain-name | name-server ip-address-of-name-server }
no ip dns { domain-name domain-name | name-server ip_address_of_name_server }
```

## Parameters

**domain-name** *domain-name*

Specifies the DNS domain name.

**name-server** *ip-address-of-name-server*

Specifies the IP address of the name server. IPv6 and IPv4 addresses are supported.

## Modes

Global configuration mode

## Usage Guidelines

- Your first run of **ip dns name-server** specifies the default IP gateway address. Your second run of **ip dns name-server** specifies the secondary IP gateway address.
- Name servers can only be entered or removed one at a time. The newly entered name server will append to the existing name server.
- The **no** form of the command with the domain-name parameter disables IP directed broadcasts for a specific domain.
- The **no** form of the command with the name-server parameter deletes a name server definition.

## Examples

The following example configures the DNS domain name and the primary name server IP address.

```
device(config)# ip dns domain-name mycompany.com
device(config)# ip dns name-server 10.70.20.1
```

## Related Commands

[show running-config ip dns](#)

# ip extcommunity-list

Configures a BGP extended community filter.

## Syntax

```
ip extcommunity-list number { deny | permit [ rt value | soo value ] reg-expr }
no ip extcommunity-list number
```

## Command Default

No BGP extended community filter is set.

## Parameters

*number*

Specifies an extended community list Instance number. Range is from 0 through 99 for a standard list (RT- or SOO-based), and from 100 through 500 for an expanded list (regular-expression-based).

**deny**

Denies access for a matching condition.

**permit**

Permits access for a matching condition.

**rt**

Specifies the route target (RT) extended community.

*value*

This value can be entered in one of the following formats:

- autonomous-system-number : network-number
- ip-address : network-number

**soo**

Specifies the site of origin (SOO) extended community.

*value*

This value can be entered in one of the following formats:

- autonomous-system-number : network-number
- ip-address : network-number

*reg-expr*

Specifies a regular expression. See the Usage Guidelines.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Refer to the topic "BGP regular expression pattern-matching characters" in the *Network OS Layer 3 Routing Configuration Guide*.

Use the **no** form of this command to delete a BGP extended community list.

## Examples

The following example specifies a standard extended community list and permit a route target and deny a site of origin.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip extcommunity-list 99 permit rt 123:2
device(config-rbridge-id-122)# ip extcommunity-list 99 deny soo 124:1
```

The following example specifies an expanded Extended Community list and permit a regular expression.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip extcommunity-list 101 permit 100:*
```

The following example deletes an extended community list:

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# no ip extcommunity-list 101
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.1.0	This command was modified to support regular expressions with an expanded range.

# ip fabric-virtual-gateway

Enables IPv4 Fabric-Virtual-Gateway configurations, for use with VRF address family IPv4 unicast.

## Syntax

`ip fabric-virtual-gateway`

`no ip fabric-virtual-gateway`

## Command Default

None

## Modes

Interface Ve configuration mode

## Usage Guidelines

VRF address family IPv4 unicast must be configured on a VRF instance, by means of the **address-family ipv4 unicast** command in VRF configuration mode.

Enter the **no** form of the command to disable IPv4 Fabric-Virtual-Gateway configurations.

## Examples

The following example shows how to enable IPv4 Fabric-Virtual-Gateway on global VE interface.

```
device(config)# interface ve 2000
device(config-Ve-2000)# ip fabric-virtual-gateway
device(config-ip-fabric-virtual-gw) #
```

## History

Release version	Command history
5.0.1	This command was introduced.
6.0.1	The description of this command was updated.

# ip icmp address-mask

Enables IPv4 Internet Control Message Protocol (ICMP) address masks.

## Syntax

```
ip icmp address-mask  
no icmp address-mask
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This is an interface-specific configuration that is persistent across a device reload.

To disable address masks, use the **no** form of this command.

## Examples

The following example enables IPv4 ICMP address masks on an Ethernet interface.

```
device(config)# interface fortygigabitethernet 1/0/50  
device(conf-int-fo-1/0/50)# ip icmp address-mask
```

## History

Release version	Command history
5.0.1	This command was introduced.



# ip icmp echo-reply

Enables the generation of IPv4 Internet Control Message Protocol (ICMP) Echo Reply messages.

## Syntax

```
ip icmp echo-reply
```

```
no ip icmp echo-reply
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Modes

Interface configuration mode

## Usage Guidelines

This is an interface-specific configuration that is persistent across a switch reload.

To disable Echo Reply messages, use the **no** form of this command.

## Examples

The following example enables IPv4 ICMP Echo Reply messages on an Ethernet interface.

```
device(config)# interface fortygigabitethernet 1/0/50
device(conf-int-fo-1/0/50)# ip icmp echo-reply
```

# ip icmp rate-limiting

Limits the rate at which IPv4 Internet Control Message Protocol (ICMP) messages are sent on a network.

## Syntax

```
ip icmp rate-limiting milliseconds  
no ip icmp rate-limiting
```

## Command Default

None

## Parameters

*milliseconds*

Time interval per ICMP packet in milliseconds. The range is from 0 through 4294967295. The default is 1000.

## Modes

Interface configuration mode

## Usage Guidelines

This is an interface-specific configuration.

The **no** form of the command will revert to the default setting. Set the interval to 0 to disable IPv4 ICMP rate-limiting.

## Examples

The following example enables IPv4 ICMP rate-limiting on an Ethernet interface.

```
device(config)# interface fortygigabitethernet 1/0/50  
device(conf-int-fo-1/0/50)# ip icmp rate-limiting
```

# ip icmp redirect

Enables IPv4 Internet Control Message Protocol (ICMP) Redirect messages, which request that packets be sent on an alternative route.

## Syntax

```
ip icmp redirect
no ip icmp redirect
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Modes

Interface configuration mode

## Usage Guidelines

This is an interface-specific configuration.

The **no** form of the command disables IPv4 ICMP Redirect messages.

## Examples

The following example enables IPv4 ICMP Redirect messages on an Ethernet interface.

```
device(config)# interface fortygigabitethernet 1/0/50
device(conf-int-fo-1/0/50)# ip icmp redirect
```

## History

Release version	Command history
5.0.1	This command was introduced.

# ip icmp unreachable

Prohibits routers from forwarding an IPv4 Internet Control Message Protocol (ICMP) Destination Unreachable Code 3 (port unreachable) message on a point-to-point link back onto the ingress port.

## Syntax

```
ip icmp unreachable  
no ip icmp unreachable
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Modes

Interface subtype configuration mode

## Usage Guidelines

By default, ICMP Destination Unreachable Code 3 messages are sent for a discarded IPv4 packet. Packets are trapped and a corresponding error message is returned if either the port, host, or network is unreachable.

This is an interface-specific configuration that is persistent across a device reload.

Use the **no** form of this command to disable the sending of the messages.

## Examples

The following example enables IPv4 ICMP Destination Unreachable Code 3 messages on an Ethernet interface.

```
device(config)# interface fortygigabitethernet 1/0/50  
device(conf-int-fo-1/0/50)# ip icmp unreachable
```

# ip igmp immediate-leave

Removes a group from the IGMP table immediately following receipt of a Leave Group request.

## Syntax

**ip igmp immediate-leave**

**no ip igmp immediate-leave**

## Command Default

This command is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command treats an interface as if it had one multicast client, so that the receipt of a Leave Group request on the interface causes the group to be removed immediately from the multicast database.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, GigabitEthernet, HundredGigabitEthernet, and port-channel interfaces. This command is not supported on VLANs.

Enter the **no** form of this command to restore the default behavior.

## Examples

To configure an Ethernet interface to remove a group from the IGMP table immediately following receipt of a Leave Group request:

```
device(config)# interface tengigabitethernet 1/0/1
device(config-if-te-1/0/1)# ip igmp immediate-leave
```

## History

Release version	Command history
7.0.0	This command was modified to include support for port-channel interfaces.

# ip igmp last-member-query-count

Sets the last member query count for a routed port. The last member query count is used while processing the leave message. If the routed port is a virtual Ethernet (VE) interface, the configuration is applied on the corresponding VLAN.

## Syntax

```
ip igmp last-member-query-count value
no ip igmp last-member-query-count value
```

## Command Default

See Parameters.

## Parameters

*value*  
Range is from 2 through 10. The default is 2.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The last member query count is used while processing the leave message. If the routed port is a virtual Ethernet (VE) interface, the configuration is applied on the corresponding VLAN.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, port-channel, GigabitEthernet, HundredGigabitEthernet, and VLAN interfaces.

Enter the **no** form of this command to restore the default.

## Examples

The following example sets the last member query count for a VLAN to 3.

```
switch(config)# interface vlan 100
switch(conf-vlan-100)# ip igmp last-member-query-count 3
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ip igmp last-member-query-interval

Sets the IGMP last-member query interval for an interface.

## Syntax

`ip igmp last-member-query-interval milliseconds`

`no ip igmp last-member-query-interval`

## Command Default

See Parameters.

## Parameters

*milliseconds*

Response time in milliseconds. Range is from 100 through 25500 milliseconds. The default is 1000.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The last-member query interval is the time in seconds that the IGMP router waits to receive a response to a group-specific query message, including messages sent in response to a host-leave message.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, GigabitEthernet, HundredGigabitEthernet, port-channel, and VLAN interfaces.

Enter the **no** form of this command to restore the default.

## Examples

To set the last-member query interval to 1500 milliseconds on a specific VLAN interface:

```
device(config)# interface vlan 100
device(conf-Vlan-100)# ip igmp last-member-query-interval 1500
```

## History

Release version	Command history
7.0.0	This command was modified to include support for Ethernet and port-channel interfaces.

# ip igmp query-interval

Sets the IGMP query interval for an interface.

## Syntax

```
ip igmp query-interval seconds
```

```
no ip igmp query-interval seconds
```

## Command Default

See Parameters.

## Parameters

*seconds*

Response time in seconds. Range is from 1 through 18000 seconds. The default is 125.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The query interval is the amount of time between IGMP query messages sent by the device.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, GigabitEthernet, HundredGigabitEthernet, port-channel, and VLAN interfaces.

Enter the **no** form of this command to restore the default.

## Examples

To set the query interval to 500 seconds on a specific VLAN interface:

```
device(config)# interface vlan 100
device(conf-Vlan-100)# ip igmp query-interval 500
```

To remove the query interval from a specific VLAN interface:

```
device(config)# interface vlan 100
device(conf-Vlan-100)# no ip igmp query-interval
```

## History

Release version	Command history
7.0.0	This command was modified to include support for Ethernet and port-channel interfaces.



# ip igmp query-max-response-time

Sets the maximum response time for IGMP queries for an interface.

## Syntax

```
ip igmp query-max-response-time seconds
no ip igmp query-max-response-time
```

## Command Default

See Parameters.

## Parameters

*seconds*

Response time in seconds. Range is from 1 through 25 seconds. The default is 10.

## Modes

Interface subtype configuration mode

## Usage Guidelines

When a host receives the query packet, it starts counting to a random value, less than the maximum response time. When this timer expires, the switch (host) replies with a report, provided that no other host from the same group has responded yet.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, port-channel, and VLAN interfaces.

Enter the **no** form of this command to restore the default.

## Examples

To set the maximum response time to 20 seconds on a VLAN interface:

```
device(config)# interface vlan 100
device(conf-Vlan-100)# ip igmp query-max-response-time 20
```

To restore the default maximum response time:

```
switch(config)# interface vlan 100
switch(conf-Vlan-100)# no ip igmp query-max-response-time
```

## History

Release version	Command history
7.0.0	This command was modified to include support for Ethernet and port-channel interfaces.

# ip igmp snooping enable (global version)

Enables Internet Group Management Protocol (IGMP) snooping for all VLAN interfaces.

## Syntax

**ip igmp snooping enable**

**no ip igmp snooping enable**

## Command Default

IGMP snooping is disabled globally.

## Modes

Global configuration mode

## Usage Guidelines

You must enable snooping at the global and VLAN levels. Enabling snooping at a global level does not enable snooping on any of the VLANs, however disabling snooping at the global level will disable snooping on all VLANs. This behavior is not supported on clusters where few nodes in the cluster are running NOS firmware prior to 7.0.0 and others in the cluster are running the 7.0.0 firmware.

Enter **no ip igmp snooping enable** to return to the default setting.

## Examples

To enable IGMP globally:

```
switch(config)# ip igmp snooping enable
```

## History

Release version	Command history
7.0.0	This command was modified to include changes in the IGMP snooping behavior.

## Related Commands

[ip igmp snooping enable](#), [show ip igmp snooping](#)

# ip igmp snooping enable

Enables Internet Group Management Protocol (IGMP) snooping.

## Syntax

```
ip igmp snooping enable
```

```
no ip igmp snooping enable
```

## Command Default

When snooping is enabled globally, IGMP snooping is enabled on all VLAN interfaces.

## Modes

Interface subtype configuration mode

## Usage Guidelines

IGMP snooping allows a network device to listen in on the IGMP conversation between hosts and routers. By listening to these conversations, the switch maintains a map of which links need which IP multicast streams. Multicasts may be filtered from the links which do not need them.

You must enable snooping at the global and VLAN levels. Enabling snooping at a global level does not enable snooping on any of the VLANs, however disabling snooping at the global level will disable snooping on all VLANs. This behavior is not supported on clusters where few nodes in the cluster are running NOS firmware prior to 7.0.0 and others in the cluster are running the 7.0.0 firmware.

Enter **no ip igmp snooping enable** to disable snooping for a specific VLAN interface.

## Examples

To enable IGMP globally:

```
device(config)# ip igmp snooping enable
```

To enable IGMP for a specific VLAN interface:

```
device(config)# interface vlan 1  
device(config-Vlan-1)# ip igmp snooping enable
```

To disable IGMP for a specific VLAN interface:

```
device(config)# interface vlan 1  
device(config-Vlan-1)# no ip igmp snooping enable
```

## ip igmp snooping fast-leave

Enables Internet Group Management Protocol (IGMP) snooping fast-leave processing for a VLAN. This allows the removal of an interface from the forwarding table without sending out group-specific queries to the interface.

### Syntax

**ip igmp snooping fast-leave**

**no ip igmp snooping fast-leave**

### Command Default

This command is disabled.

### Modes

Interface subtype configuration mode

### Usage Guidelines

Enter **no ip igmp snooping fast-leave** to disable this function.

### Examples

To enable snooping fast-leave for a specific VLAN interface:

```
device(config)# interface vlan 1
device(config-Vlan-1)# ip igmp snooping fast-leave
```

To disable snooping fast-leave for a specific VLAN interface:

```
device(config)# interface vlan 1
device(config-Vlan-1)# no ip igmp snooping fast-leave
```

# ip igmp snooping mrouter interface

Configures a VLAN port member to be a multicast router interface.

## Syntax

```
ip igmp snooping mrouter interface { <N>gigabitethernet rbridge-id/slot/port | port-channel number | tunnel value}
```

```
no ip igmp snooping mrouter interface { <N>gigabitethernet rbridge-id/slot/port | port-channel number | tunnel value}}
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

#### port-channel*interface number*

Specifies the interface is a port-channel. Valid values range from 1 through 6144.

#### tunnel*value*

Specifies the interface is a tunnel. Valid values range from 1 through 100000.

## Modes

Interface subtype configuration mode

## Usage Guidelines

A multicast router interface faces toward a multicast router or other Internet Group Management Protocol (IGMP) querier.

The **no** form of this command removes the configured mrouter.

## Examples

The following example configures a VLAN port member to be a multicast router interface.

```
device(config)# interface vlan 100
device(config-vlan-100)# ip igmp snooping mrouter interface tengigabitethernet 101/0/1
```

# ip igmp snooping querier enable

Activates or deactivates the Internet Group Management Protocol (IGMP) snooping querier on a VLAN.

## Syntax

```
ip igmp snooping querier enable  
no ip igmp snooping querier enable
```

## Command Default

IGMP snooping querier is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no ip igmp snooping querier enable** to disable the IGMP snooping querier.

## Examples

To enable the IGMP snooping querier feature for the VLAN interface:

```
device(config)# interface vlan 100  
device(config-vlan-100)# ip igmp snooping querier enable
```

# ip igmp snooping restrict-unknown-multicast

Stops the flooding of unknown multicast traffic in a VLAN domain.

## Syntax

```
ip igmp snooping restrict-unknown-multicast
no ip igmp snooping restrict-unknown-multicast
```

## Command Default

IGMP snooping restrict-unknown-multicast is enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The hardware profile **ipv4-v6-mcast** must be enabled, by means of the **hardware-profile** command.

Using the **no ip igmp snooping restrict-unknown-multicast** command will flood multicast traffic to all members of the VLAN.

## Examples

To stop the flooding of unknown multicast traffic for a VLAN enter the following commands:

```
switch(config)# interface vlan 100
switch(config-vlan-100)# ip igmp snooping restrict-unknown-multicast
```

## History

Release version	Command history
7.0.0	This command was modified to indicate the requirement that <b>ipv4-v6-mcast</b> be enabled.

# ip igmp snooping robustness-variable

Configures a value to compensate for IGMP snooping packet loss in congested networks.

## Syntax

`ip igmp snooping robustness-variable value`

`no ip igmp snooping robustness-variable value`

## Command Default

See Parameters.

## Parameters

*value*

The number of general IGMP snooping queries sent before a multicast address is aged out. The range is from 2 through 10. The default is 2.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This value determines the number of general IGMP snooping queries that are sent before a multicast address is aged out for lack of a response. Use this command to configure the robustness variable on an interface.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, port-channel, and VLAN interfaces.

The **no** form of the command restores the robustness variable value to 2 (the default).

## Examples

The following example changes the robustness variable on a VLAN to 7.

```
device(config)# interface vlan 2000
device(config-vlan-2000)# ip igmp snooping robustness-variable 7
```

## History

Release version	Command history
7.0.0	This command was introduced.



# ip igmp startup-query-count

Sets the IGMP startup query count for an interface.

## Syntax

```
ip igmp startup-query-count value
```

```
no ip igmp startup-query-count value
```

## Command Default

See Parameters.

## Parameters

*value*

The number of queries sent at startup. The range is from 1 through 10. The default is 2.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command is useful when the IGMP querier starts the first time.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, GigabitEthernet, HundredGigabitEthernet, port-channel, and VLAN interfaces.

Use the **no** form of this command to restore the default.

## Examples

The following example changes the IGMP startup query count on a VLAN from the default to 3.

```
device(config)# interface vlan 100
device(conf-vlan-100)# ip igmp startup-query-count 3
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ip igmp startup-query-interval

Sets the IGMP startup query interval for an interface.

## Syntax

```
ip igmp startup-query-interval seconds
```

```
no ip igmp startup-query-interval seconds
```

## Command Default

See Parameters.

## Parameters

*seconds*

The response time in seconds. Range is from 1 through 450. The default is 31.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command is useful when the IGMP querier starts the first time.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, GigabitEthernet, HundredGigabitEthernet, port-channel, and VLAN interfaces.

The **no** form of the command restores the startup query interval to the default.

## Examples

The following example sets the IGMP startup query interval for a VLAN to 200 seconds.

```
device(config)# interface vlan 100
device(conf-vlan-100)# ip igmp startup-query-interval 200
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ip igmp static-group

Configures the IGMP static group membership entries for a specific interface.

## Syntax

```
ip igmp static-group A.B.C.D  
no ip igmp static-group A.B.C.D
```

## Parameters

*A.B.C.D*

Specifies the group address, as a subnet number in dotted decimal format (for example, 10.0.0.1), as the allowable range of addresses to be included in the multicast group.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **ip igmp static-group** command creates IGMP static group membership to test multicast forwarding without a receiver host. Traffic is forwarded to an interface without the need to receive membership reports from host members. Packets to the group are fast-switched out of a specific interface. Static group membership entries are automatically added to the IGMP cache and the PIM mcache table.

This command is supported on TenGigabitEthernet, FortyGigabitEthernet, port-channel, and VLAN interfaces.

## Examples

To create a static multicast group for a VLAN interface:

```
device(config)# interface vlan 100  
device(config-Vlan-100)# ip igmp static-group 225.1.1.1 interface port-channel 60
```

To clear a static group on a VLAN interface:

```
device(config)# interface vlan 100  
device(config-Vlan-100)# no ip igmp static-group 225.1.1.1
```

To create a static multicast group for a routed port:

```
device(config)# interface tengigabitethernet 58/0/7  
device(conf-if-te-58/0/7)# ip igmp static-group 225.0.0.10
```

To clear the static multicast group enter the following command.

```
device(config)# interface tengigabitethernet 58/0/7  
device(conf-if-te-58/0/7)# no ip igmp static-group 225.0.0.10
```

## History

Release version	Command history
7.0.0	This command was modified to include support for Ethernet and port-channel interfaces.

# ip import routes (IPv4 VRF address-family configuration mode)

Leaks IPv4 routes from one VRF to the VRF you are configuring, based on match criteria defined in route-map.

## Syntax

```
ip import routes VRF_name route-map rmap_name
```

```
no ip import routes
```

## Parameters

*VRF\_name*

Specifies the VRF instance from which to leak routes to the VRF you are configuring.

*rmap\_name*

Specifies the name of route map to use for route-leaking match criteria. Range is from 1 through 63 ASCII characters.

## Modes

IPv4 VRF address-family configuration mode

## Usage Guidelines

Use the **no** form of the command to remove routes from being leaked to the VRF you are configuring.

## Examples

To leak IPv4 routes from a VRF named "red" to the VRF named "orange," based on match criteria from a route map named "import-map," with an example RBridge ID of 10:

```
switch# configure
switch(config)# rbridge-id 10
switch(config-rbridge-id-10)# vrf orange
switch(config-vrf-orange)# address-family ipv4 unicast
switch(config-ipv4-unicast)# ip import routes red route-map import-map
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The keyword <b>map</b> was corrected to <b>route-map</b> .

# ip import routes (RBridge ID configuration mode)

Leaks IPv4 routes from the specified VRF to the default VRF, based on match criteria defined in route-map.

## Syntax

```
ip import routes VRF_name map rmap_name  
no ip import routes
```

## Parameters

*VRF\_name*  
Specifies the VRF instance from which to leak routes to the default VRF.

*rmap\_name*  
Specifies the map name to use for route-leaking match criteria.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no** form of the command to remove routes from being leaked to the default VRF.

## Examples

To leak IPv4 routes from a VRF named "red" to the default VRF, based on match criteria from a route map named "import-map" :

```
switch# configure  
switch(config)# rbridge-id 54  
switch(config-rbridge-id-54)# ip import routes red map import-map
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ip interface

Sets the IP address of the VXLAN overlay gateway instance.

## Syntax

```
ip interface { veveid vrrp-extended-group group-ID | loopback ifid }
no ip interface
```

## Parameters

**ve** *veid*

Specifies the ID of the virtual Ethernet (VE) interface (which must already be configured) through which you are configuring the IP address of the VXLAN gateway.

**vrrp-extended-group** *group-ID*

Specifies the virtual router group (which must already be configured) through which you are configuring the IP address of the VXLAN gateway.

**loopback** *ifid*

Specifies an IPv4 loopback interface ID (IPv6 addresses are ignored). The range is from 1 through 255.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

### ve vrrp-extended-group

The VXLAN overlay gateway IP address is also the source IP address for all the tunnels associated with the gateway. The command accepts the VE interface ID and VRPP-E group ID, then sets the overlay gateway IP address to be identical to the already configured Virtual Redundancy Router Protocol-Extended (VRRP-E) virtual IP address.

Ensure that the VXLAN gateway is in the inactive state when you issue this command. Use the **no activate** command in VXLAN overlay gateway configuration mode to deactivate the gateway.

If you have already added RBridge attachments to the VXLAN gateway overlay, the VE and VRPP-E group IDs must exist for the attached RBridge IDs.

Changing the VE interface ID or VRRP-E group ID requires an update of all tunnel source addresses.

Some commands cannot be used if they would affect an active VXLAN gateway address configuration. For example, consider the following configuration:

```
switch# configure
switch(config)# overlay-gateway GW1
switch(config-overlay-gw-GW1)# attach rbridge-id add 1
switch(config-overlay-gw-GW1)# ip interface ve 1000 vrrp-extended-group 100
switch(config-overlay-gw-GW1)# activate
```

Examples of operations that would not be allowed based on this configuration are the following:

- Deleting VLAN 1000 (because this implicitly deletes VE 1000)

- Deleting VE 1000 on RBridge 1
- Deleting VRRP-E group 100 for VE 1000 on RBridge 1
- Changing virtual IP configuration for VE 1000, VRRPE group 100 on RBridge 1
- Changing the VRF on VE 1000 on RBridge 1

### loopback

If a specified loopback ID does not exist, or if the loopback interface is not fully configured, the **activate** command is rejected.

Use the **no** form of this command to delete the IP address configuration for this gateway.

## Examples

To set the IP address of a VXLAN gateway overlay named "GW1" (using the already configured VE interface ID 10 and the vrrp-extended group ID 25):

```
switch# configure
switch(config)# overlay-gateway GW1
switch(config-overlay-gw-GW1)# ip interface ve 10 vrrp-extended-group 25
```

## History

Release version	Command history
6.0.0	This command was modified to support distributed VXLAN gateways.



# ip mtu

Sets the IP maximum transmission unit (MTU) globally or on an interface.

## Syntax

```
ip mtu size
```

```
no ip mtu
```

## Command Default

The default IP MTU size is 1500 bytes.

## Parameters

*size*

Specifies the size of an interface IP MTU. Values range from 1300 through 9000 bytes.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The **no** form of the command reverts the MTU size to the default value.

The entire fabric acts like a single switch. Therefore, IP MTU is applicable only on edge ports and not on an ISL.

If the interface is part of a VE, change the IPv4 MTU only at the VE interface and not at the physical port. All member ports of a VE inherit the VE-interface IPv4 MTU value.

## Examples

The following example sets the IP MTU to 2000 bytes on the specified Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(config-if-te-178/0/9)# ip mtu 2000
```

The following example changes the IP MTU for a VE.

```
device# configure terminal
device(config)# interface ve 103
device(config-vif-103)# ip mtu 2000
```

The following example changes the IP MTU globally.

```
device(config)# ip mtu 2000
```

## Related Commands

[interface](#)

# ip multicast-boundary

Configures a multicast boundary on an interface. You can also filter a range of multicast-group addresses by specifying a prefix list.

## Syntax

```
ip multicast-boundary [ prefix-list ]  
no ip multicast-boundary
```

## Command Default

No multicast boundaries are defined on an interface.

## Parameters

*prefix-list*

Specifies the name of a prefix list defined by the **ip prefix-list** command. Permitted values are between 1 and 63 characters. Although the first character must be alphabetic, the others can be alphanumeric, underscores (\_), or minus signs (-).

## Modes

Interface subtype configuration mode

## Usage Guidelines

If a *prefix-list* is not specified, this command applies the boundary for the entire multicast range on the interface.

If a *prefix-list* is specified, this command considers only multicast-IP rules. Unicast-IP rules are ignored.

To disable this feature, enter **no ip multicast-boundary**.

## Examples

The following example sets the multicast boundary on an interface, specifying a prefix list. This interface acts as a boundary: all communication across this interface will be discarded for a multicast group which matches the rules in the provided *prefix-list*.

```
device# configure terminal  
device(config)# interface tengigabitethernet 1/3/1  
device(conf-if-te-1/3/1)# ip multicast-boundary abc  
device(conf-if-te-1/3/1)#
```

## Related Commands

[ip prefix-list](#), [show ip pim-sparse](#)

# ip ospf active

Sets a specific OSPF interface to active.

## Syntax

```
ip ospf active
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **ip ospf active** command on each interface participating in adjacency formation. This command overrides the global passive setting on that interface, and enables transmission of OSPF control packets.

## Examples

The following example sets a specific OSPFv2 virtual Ethernet (VE) interface to active.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# interface ve 100
device(config-Ve-100)# ip ospf active
```

## Related Commands

[ip ospf passive](#)

# ip ospf area

Enables OSPFv2 on an interface.

## Syntax

```
ip ospf area area-id | ip-addr
no ip ospf area
```

## Command Default

Disabled.

## Parameters

*area-id*  
Area ID in decimal format. Valid values range from 1 through 2147483647.

*ip-addr*  
Area ID in IP address format.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command disables OSPFv2 on the interface.

## Examples

The following example enables a configured OSPFv2 area named 0 on a specific OSPFv3 10-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/49
device(conf-if-te-1/0/49)# ip ospf area 0
```

The following example enables a configured OSPFv2 area named 0 on a specific OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 177
device(config-rbridge-id-177)# interface ve 12
device(config-Ve-12)# ip ospf area 0
```

# ip ospf auth-change-wait-time

Configures authentication-change hold time.

## Syntax

```
ip ospf auth-change-wait-time wait-time  
no ip ospf auth-change-wait-time
```

## Command Default

Wait time is 300 seconds

## Parameters

*wait-time*

Time before an authentication change takes place. Valid values range from 0 to 14400 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the authentication change hold time for the interface to which you are connected.

OSPFv2 provides graceful authentication change for the following types of authentication changes:

Changing authentication methods from one of the following to another of the following:

- Simple text password
- MD5 authentication
- No authentication

Configuring a new simple text password or MD5 authentication key.

Changing an existing simple text password or MD5 authentication key

The **no** form of the command resets the wait time to the default of 300 seconds.

## Examples

The following example sets the wait time to 600 seconds on a specific OSPFv2 10-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface tengigabitethernet 190/0/49  
device(conf-if-te-190/0/49)# ip ospf auth-change-wait-time 600
```

The following example sets the wait time to 400 seconds on a specific OSPF virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 12
device(config-ve-12)# ip ospf auth-change-wait-time 400
```

# ip ospf authentication-key

Configures simple password-based authentication for OSPF.

## Syntax

```
ip ospf authentication-key { 0 password | 2 password | 255 password | password }
no ip ospf authentication-key
```

## Command Default

Authentication is disabled.

## Parameters

*0 password*

No encryption. OSPF processes *password* as a plain text password and shows the unencrypted password in the **show running** command output as follows: `key 0 passwd`

*2 password*

Expects the user to provide the encrypted password, preceded by a dollar sign (\$) sign, and shows the encrypted password in the **show running** command output as follows: `key 2 $c1pVT0=`

*255 password*

Expects the user to provide the encrypted password, and **255** internally maps to **2**. OSPF shows the encrypted password in the **show running** command output as follows: `key 2 $c1pVT0=`

*password*

OSPF processes *password* as a plain text password. OSPF internally encrypts this password as if encryption key 2 was specified and shows the encrypted password in the **show running** command output as follows: `key 2 $c1pVT0=`

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset simple password-based authentication on the OSPFv2 interface to which you are connected. The **no** form of the command disables OSPFv2 authentication.

## Examples

The following example sets authentication only on the specified OSPFv2 10-gigabit Ethernet interface. A plain text password called `extreme` must be entered that OSPFv2 will encrypt as if encryption key 2 was specified.

```
device# configure terminal
device(config)# interface tengigabitethernet 190/0/49
device(conf-if-te-190/0/49)# ip ospf authentication-key extreme
```



The following example sets authentication on the OSPFv2 virtual Ethernet (VE) interface 12, with a plain text password called extreme that OSPF will encrypt as if encryption key 2 was specified.

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 12
device(config-Ve-12)# ip ospf authentication-key extreme
```

# ip ospf bfd

Enables Bidirectional Forwarding Detection (BFD) on a specific OSPFv2 interface.

## Syntax

```
ip ospf bfd
no ip ospf bfd
```

## Command Default

BFD is disabled by default.

## Modes

Interface subtype configuration mode

## Usage Guidelines

BFD sessions are initiated only if BFD is also enabled globally using the `bfd` command in OSPF router configuration mode. If BFD is disabled using the `no bfd` command in OSPF router configuration mode, BFD sessions on specific OSPFv2 interfaces are deregistered.

The `no` form of the command removes all BFD sessions from a specified interface.

## Examples

The following example enables BFD on an OSPF 10-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 101/0/10
device(conf-if-te-101/0/10)# ip ospf bfd
```

The following example disables BFD on an OSPF virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 24
device(config-ve-24)# no ip ospf bfd
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ip ospf cost

Configures cost for a specific interface.

## Syntax

```
ip ospf cost value  
no ip ospf cost
```

## Command Default

Cost value is 1.

## Parameters

*value*

Cost value. Valid values range from 1 through 65535. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the OSPFv2 cost on the interface. If the cost is not configured with this command, OSPFv2 calculates the value from the reference and interface bandwidths.

The **no** form of the command disables the configured cost.

## Examples

The following example sets the cost to 600 on a specific OSPFv2 10-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface tengigabitethernet 190/0/49  
device(config-if-te-190/0/49)# ip ospf cost 600
```

The following example sets the cost to 520 on a specific OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal  
device(config)# rbridge-id 178  
device(config-rbridge-id-178)# interface ve 12  
device(config-Ve-12)# ip ospf cost 520
```

# ip ospf database-filter

Configures filters for different types of outgoing Link State Advertisements (LSAs).

## Syntax

```
ip ospf database-filter { all-external | all-summary-external { allow-default-and-type-4 | allow-default-out | out } }
```

```
ip ospf database-filter all-out
```

```
no ip ospf database-filter all-external
```

```
no ip ospf database-filter all-out
```

```
no ip ospf database-filter all-summary-external
```

## Command Default

All filters are disabled.

## Parameters

### all-external

Blocks all external LSAs.

### all-summary-external

Blocks all summary (Type 3) and external (type 5) LSAs.

### allow-default-and-type-4

Allows default-route LSAs and Type 4 LSAs, but block all other LSAs.

### allow-default-out

Allows default-route LSAs, but block all other LSAs.

### out

Filters outgoing LSAs.

### all-out

Blocks all LSAs.

## Modes

Interface subtype configuration mode

## Usage Guidelines

By default, the device floods all outbound LSAs on all the OSPF interfaces within an area. You can configure a filter to block outbound LSAs on an OSPF interface. This feature is particularly useful when you want to block LSAs from some, but not all, of the interfaces attached to the area. When enabled, this command blocks the specified outgoing LSAs on the interface. Some cases where you might want to enable filters are:

- To control the information being advertised to the network.
- To use a passive router for debugging only.

The **no** form of the command disables configurations.

**NOTE**

You cannot block LSAs on virtual links.

## Examples

The following example applies a filter to block flooding of all LSAs on a specific OSPF 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/10
device(conf-if-fo-101/0/10)# ip ospf database-filter all-out
```

# ip ospf dead-interval

Configures the neighbor dead interval, which is the number of seconds that a neighbor router waits for a hello packet from the device before declaring the router down.

## Syntax

```
ip ospf dead-interval interval
```

```
no ip ospf dead-interval
```

## Command Default

The specified time period is 40 seconds.

## Parameters

*interval*

Dead interval in seconds. Valid values range from 3 through 65535 seconds. The default is 40.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the dead interval, the hello interval is automatically changed to a value that is one fourth that of the new dead interval, unless the hello interval is also explicitly configured using the **ip ospf hello-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is  $\frac{1}{4}$  times that of the dead interval.

The **running-config** command displays only explicitly configured values of the hello interval, which means that a value that was automatically changed as the result of a dead-interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the dead interval to 200 on a specific OSPFv2 40-gigabit Ethernet interface.

```
device# device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface fortygigabitethernet 101/0/10
device(conf-if-fo-101/0/10)# ip ospf dead-interval 200
```

The following example sets the dead interval to 200 on a specific OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 24
device(config-Ve-24)# ip ospf dead-interval 200
```

# ip ospf hello-interval

Configures the hello interval, which is the length of time between the transmission of hello packets that this interface sends to neighbor routers.

## Syntax

```
ip ospf hello-interval interval
```

```
no ip ospf hello-interval
```

## Command Default

The default value is 10 seconds.

## Parameters

*interval*

Hello interval in seconds. Valid values range from 1 through 65535.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the hello interval, the dead interval is automatically changed to a value that is four times that of the new hello interval, unless the dead interval is also explicitly configured using the **ip ospf dead-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is  $\frac{1}{4}$  times that of the dead interval.

The **running-config** command displays only explicitly configured values of the dead interval, which means that a value that was automatically changed as the result of a hello-interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the hello interval to 50 on a specific OSPFv2 40-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/10
device(conf-if-fo-101/0/10)# ip ospf hello-interval 50
```

The following example sets the hello interval to 50 on a specific OSPFv2 virtual Ethernet (VE) interface:

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 24
device(config-Ve-24)# ip ospf hello-interval 50
```



# ip ospf md5-authentication

Configures MD5 password and authentication change hold time.

## Syntax

```
ip ospf md5-authentication { key-activation-wait-time wait-time | key-id id MD5_key { 0 | 2 | 255 } ospf_password }
no ip ospf md5-authentication key-id
```

## Command Default

No authentication.

## Parameters

### key-activation-wait-time

Sets the time that OSPF waits before activating a new key.

*wait-time*

Time OSPF waits before activating a new MD5 key. This parameter provides a graceful transition from one MD5 key to another without disturbing the network. All new packets transmitted after the wait time ends will use the newly configured MD5 Key. OSPF packets that contain the old MD5 key are accepted for up to five minutes after the new MD5 key is in operation. Valid values range from 0 to 14400 seconds. The default value is 300 seconds.

### key-id

Sets MD5 key and OSPF password.

*id MD5\_key*

The *num* is a number between 1 and 255 and identifies the MD5 key that is being used. This parameter is required to differentiate among multiple keys defined on a router. When MD5 is enabled, the *key* is an alphanumeric password of up to 16 characters that is later encrypted and included in each OSPF packet transmitted. You must enter a password in this field when the system is configured to operate with either simple or MD5 authentication. By default, the MD5 authentication key is encrypted.

**0** *password*

No encryption. OSPF processes **password** as a plain text password and shows the unencrypted password in the **show running** command output as follows: `key 0 passwd`

**2** *password*

Expects the user to provide the encrypted password, preceded by a dollar sign (\$), and shows the encrypted password in the **show running** command output as follows: `key 2 $c1pVT0=`

**255** *password*

Expects the user to provide the encrypted password, and **255** internally maps to **2**. OSPF shows the encrypted password in the **show running** command output as follows: `key 2 $c1pVT0=`

*ospf\_password*

OSPF processes *password* as a plain text password. OSPF internally encrypts this password as if encryption key 2 was specified and shows the encrypted password in the **show running** command output as follows: `key 2 $c1pVT0=`

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the MD5 password and/or authentication change hold time on the interface to which you are connected.

Enter **no ip ospf md5-authentication key-id** to disable this configuration.

## Examples

The following command sets authentication only on the OSPF 40-gigabit Ethernet interface 100/0/1. To enter an MD5 ID/key of **255 key** and a plain text OSPF password called extreme that OSPF will encrypt as if encryption key **2** was specified:

```
device(config)# interface fortygigabitethernet 100/0/1
device(conf-if-fo-100/0/1)# ip ospf md5 key-id 255 key extreme
```

The following command sets authentication only on the OSPF virtual Ethernet (VE) interface 24. To enter an MD5 id/key of **255 key** and a plain text OSPF password called extreme that OSPF will encrypt as if encryption key **2** was specified:

```
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 24
device(config-Ve-24)# ip ospf md5 key-id 255 key extreme
```

# ip ospf mtu-ignore

Enables or disables maximum transmission unit (MTU) match checking.

## Syntax

```
ip ospf mtu-ignore  
no ip ospf mtu-ignore
```

## Command Default

Enabled

## Modes

Interface subtype configuration mode

## Usage Guidelines

In default operation, the IP MTU on both sides of an OSPFv2 link must be the same, and a check of the MTU is performed when Hello packets are first exchanged.

The **no** form of the command disables MTU-match checking on a specific interface.

## Examples

The following example disables MTU-match checking on a specific OSPFv2 40-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface fortygigabitethernet 101/0/10  
device(conf-if-fo-101/0/10)# no ip ospf mtu-ignore
```

The following example enables MTU-match checking on a specific OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal  
device(config)# rbridge-id 177  
device(config-rbridge-id-177)# interface ve 24  
device(config-Ve-24)# ip ospf mtu-ignore
```

# ip ospf network

Configures the network type for the interface. Point-to-point can support unnumbered links, which requires less processing by OSPF.

## Syntax

```
ip ospf network { broadcast | point-to-point }  
no ip ospf network
```

## Parameters

### broadcast

Network type is broadcast, such as Ethernet.

### point-to-point

Network type is point-to-point.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no ip ospf network** to remove the network-type configuration.

### NOTE

The network type non-broadcast is not supported at this time.

## Examples

To configure an OSPF point-to-point link on the OSPF 10-gigabit Ethernet interface whose rbridge-ID/slot/port format is 190/0/49:

```
device(config)# interface tengigabitethernet 190/0/49  
device(conf-if-te-190/0/49)# ip ospf network point-to-point
```

To configure an OSPF broadcast link on the OSPF virtual Ethernet (VE) interface 24:

```
device(config)# rbridge-id 178  
device(config-rbridge-id-178)# interface ve 24  
device(config-Ve-24)# ip ospf network broadcast
```

# ip ospf passive

Sets a specific OSPFv2 interface to passive.

## Syntax

```
ip ospf passive
```

```
no ip ospf passive
```

## Command Default

All OSPF interfaces are active.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Passive interfaces accept and process all OSPF protocol traffic, but they do not send any traffic.

You might want to set an interface to passive mode if:

- You are planning to use the router mostly for debugging purposes.
- The router is a stub and does not route traffic.

The **no** form of the command sets an interface back to active.

## Examples

The following example sets a specific OSPFv2 10-gigabit Ethernet interface to passive.

```
device# configure terminal
device(config)# interface tengigabitethernet 190/0/49
device(conf-if-te-190/0/49)# ip ospf passive
```

The following example sets a specific OSPFv2 virtual Ethernet (VE) interface to passive.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 20
device(config-Ve-20)# ip ospf passive
```

## Related Commands

[ip ospf active](#)

# ip ospf priority

Configures priority for designated router (DR) election.

## Syntax

```
ip ospf priority value
```

```
no ip ospf priority
```

## Command Default

The default value is 1.

## Parameters

*value*

Priority value. Valid values range from 0 through 255.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The OSPFv2 router assigned the highest priority becomes the designated router, and the OSPFv2 router with the second-highest priority becomes the backup router.

The **no** form of the command restores the default value.

## Examples

The following example sets a priority of 10 for the OSPFv2 router that is connected to an OSPFv2 10-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 190/0/49
device(conf-if-te-190/0/49)# ip ospf priority 10
```

The following example sets a priority of 4 for the OSPFv3 router that is connected to an OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 24
device(config-Ve-24)# ip ospf priority 10
```

# ip ospf retransmit-interval

Configures the retransmit interval. The retransmit interval is the time between Link-State Advertisement (LSA) retransmissions to adjacent routers for a given interface.

## Syntax

```
ip ospf retransmit-interval interval
```

```
no ip ospf retransmit-interval
```

## Command Default

The interval is 5 seconds.

## Parameters

*interval*

Retransmit interval in seconds. Valid values range from 0 through 3600 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command resets the retransmit interval to its default.

## Examples

The following example sets the retransmit interval to 8 for all OSPFv2 devices on an OSPFv2 10-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/49
device(config-if-te-1/0/49)# ip ospf retransmit-interval 8
```

The following example sets the retransmit interval to 26 for all OSPFv2 devices on an OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 22
device(config-Ve-22)# ip ospf retransmit-interval 26
```

# ip ospf transmit-delay

Configures transmit delay for link-update packets. The transmit delay is the estimated time required for OSPFv2 to send link-state update packets on the interface to which you are connected.

## Syntax

```
ip ospf transmit-delay value
```

```
no ip ospf transmit-delay
```

## Command Default

The transmit delay is set to 1 second.

## Parameters

*value*

Transmit delay in seconds. Valid values range from 0 through 3600 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

## Examples

The following example sets a transmit delay of 25 seconds for devices on a specific OSPFv2 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/0/49
device(config-if-te-1/0/49)# ip ospf transmit-delay 25
```

The following set a transmit delay of 45 seconds for routers on a specific OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 177
device(config-rbridge-id-177)# interface ve 22
device(config-Ve-22)# ipv6 ospf transmit-delay 43
```



# ip pim dr-priority

Configures the designated router (DR) priority of a protocol Independent Multicast (PIM) enabled interface.

## Syntax

```
ip pim dr-priority priority-value  
no ip pim dr-priority
```

## Command Default

DR priority value is 1.

## Parameters

*priority-value*  
The DR priority value. Valid values range from 0 through 65535.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no ip pim dr-priority** to disable this feature.

## Examples

To set the priority to 100 on a virtual Ethernet (VE) interface:

```
device(config)# rbridge-id 178  
device(config-rbridge-id-178)# interface ve 24  
device(config-Ve-24)# ip pim dr-priority 100
```

## Related Commands

[router pim](#)

# ip pim multinet enable

Enables or disables the multinet functionality on PIM-enabled interfaces.

## Syntax

`ip pim multinet enable`  
`no ip pim multinet enable`

## Modes

Interface subtype configuration mode

## Usage Guidelines

The `no` form of this command disables the IP PIM multinet functionality.

## Examples

The following example enables PIM multinet on 10-gigabit Ethernet interfaces.

```
device# configure terminal
device(config)# interface te 1/2/8
device(conf-if-te-1/2/8)# ip pim multinet enable
```

## History

Release version	Command history
7.1.0	This command was introduced.

# ip pim neighbor-filter

By default, directly connected routers under protocol-independent multicast (PIM) form neighborhood with one another. Using this command, you can block specified routers from neighborhood.

## Syntax

```
ip pim neighbor-filter prefix-list
```

```
no ip pim neighbor-filter
```

## Command Default

Neighbor filter is not active.

## Parameters

*prefix-list*

Specifies the name of a prefix list defined by the **ip prefix-list** command. Permitted values are between 1 and 63 characters. Although the first character must be alphabetic, the others can be alphanumeric, underscores (\_), or minus signs (-).

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of this command removes a neighbor filter.

In a prefix list applied by this command, a **deny** statement blocks specified devices from neighborhood.

In a prefix list applied by this command, a **permit** statement allows establishment of PIM neighborhood with specified unicast IP addresses. Multicast prefix-list rules have no effect on PIM neighborhood. This command is supported on port-channel, physical and VE interfaces.

## Examples

The following example sets the PIM neighbor-filter on an Ethernet interface, specifying a prefix list. This interface will form neighborhood only with those neighbors whose IP address are permitted by the rules in the provided prefix-list.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/3/1
device(conf-if-te-1/3/1)# ip pim neighbor-filter abc
device(conf-if-te-1/3/1)#
```

## History

Release version	Command history
6.0.1	This command was introduced.

## Related Commands

[ip prefix-list](#), [show ip pim neighbor](#), [show ip pim-sparse](#)

# ip pim-sparse

Enables or disables Protocol Independent Multicast Sparse Mode on port channels, physical or VE interfaces.

## Syntax

```
ip pim-sparse
```

```
no ip pim-sparse
```

## Command Default

Protocol Independent Multicast (PIM) is not enabled on an interface.

## Modes

Interface subtype configuration mode

## Usage Guidelines

PIM must be enabled on the device before enabling PIM-sparse. PIM-sparse can be enabled on interfaces

Enter **no ip pim-sparse** to disable this feature.

## Examples

To enable PIM Sparse Mode on a virtual Ethernet (VE) interface:

```
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 24
device(config-Ve-24)# ip pim-sparse
```

## Related Commands

[router pim](#)

# ip policy route-map

Enables policy-based routing (PBR) on any Layer 3 interface after ACLs and route map entries are configured.

## Syntax

```
ip policy route-map map-tag  
no ip policy route-map map-tag
```

## Parameters

*map-tag*  
The name of the route-map when it was created.

## Modes

Privileged EXEC mode

## Usage Guidelines

Enter `no ip policy route-map` to disable this feature.

## Examples

To enable policy-based routing:

```
device# configure terminal  
device(config)# ip policy route-map
```

## Related Commands

[show route-map](#), [show route-map interface](#)

# ip prefix-list

Configures an IP prefix-list instance.

## Syntax

```
ip prefix-list name [ [ deny ip-prefix/prefix-length | permit ip-prefix/prefix-length ] ge ge-value [ le le-value ] ] | seq sequence-number }
```

```
no ip prefix-list name
```

## Parameters

*name*

Permitted values are between 1 and 32 characters. Although the first character must be alphabetic, the others can be alphanumeric, underscores (\_) or minus signs (-).

**deny** *ip-prefix/prefix-length*

Denies a packet that contains a route specified in the prefix list. The prefix list matches only on the specified prefix/prefix length, unless you use the **ge** *ge-value* or **le** *le-value* parameters.

**permit** *ip-prefix/prefix-length*

Permits a packet that contains a route specified in the prefix list. The prefix list matches only on the specified prefix/prefix length, unless you use the **ge** *ge-value* or **le** *le-value* parameters.

**ge** *ge-value*

If you specify only **ge** *ge-value*, then the range is from *ge-value* to 32.

**le-value** *le-value*

If you specify only **le** *le-value*, then the range is from *le-value* to the *prefix-length* parameter.

**seq** *sequence-number*

Specifies an IPv4 prefix list sequence number. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with prefix list entry 5. The Extreme device interprets the prefix list entries in numerical order, beginning with the lowest sequence number.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Enter **no ip prefix-list** *name* to disable this feature.

The *ge-value* or *le-value* you specify must meet the following condition for *prefix-length*:

```
ge-value <= le-value <= 32
```

If you do not specify *le-value* **ge** *ge-value* or **le** *le-value*, the prefix list matches only on the exact prefix you specify with the *ip-prefix/prefix-length* parameter.

## Examples

This example denies routes on 1.2.0.0/8, where the subnet mask length must be greater than or equal to 20 and less than or equal to 28, and permits routes on 10.1.0.0/16.

```
device# config
device(config-rbridge-id-1)#
device(config-rbridge-id-1)# ip prefix-list test deny 10.0.0.0/8 ge 20 le 28
device(config-rbridge-id-1)# ip prefix-list test permit 10.1.0.0/16
```



# ip proxy-arp

Enables Proxy Address Resolution Protocol (APR) on an interface.

## Syntax

```
ip proxy-arp
```

```
no ip proxy-arp
```

## Command Default

Proxy ARP is enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Proxy ARP enables a device to answer ARP requests from devices in one network on behalf of devices in another network. Because ARP requests are MAC-layer broadcasts, they reach only the devices that are directly connected to the sender of the ARP request. Therefore, ARP requests do not cross routers.

The **no** form of the command disables Proxy ARP on an interface.

## Examples

The following example enables Proxy ARP on a specified interface.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# ip proxy-arp
```

The following example disables Proxy ARP on a specified interface.

```
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# no ip proxy-arp
```

## Related Commands

[interface](#), [interface ve](#)

# ip receive access-group

Applies an IPv4 access control list (ACL) at RBridge level. Such *receive-path* ACLs filter incoming route-processor traffic according to rules that you create.

## Syntax

```
ip receive access-group acl-name in
no ip receive access-group acl-name in
```

## Command Default

No receive-path ACLs are applied to the RBridge.

## Parameters

*acl-name*  
Specifies the name of the standard or extended IP access list.

**in**  
Specifies ingress traffic.

## Modes

RBridge ID configuration mode

## Usage Guidelines

For both interface ACLs and receive-path ACLs, you use identical commands to create the ACLs. You also use identical commands to define permit/deny rules in the ACLs. The only variance is the command you use to apply the ACL:

- To apply an interface ACL, from an interface-subtype configuration mode you use the { **ip | ipv6 | mac** } **access-group** command.
- To apply a receive-path ACL, from RBridge ID configuration mode you use the { **ip | ipv6** } **receive access-group** command.

You can apply a receive-path ACL to multiple RBridges.

You can apply a maximum of two receive-path ACLs to an RBridge, as follows:

- One IPv4 receive-path ACL
- One IPv6 receive-path ACL

To remove a receive-path ACL from an RBridge, enter the **no** form of this command.

## Examples

The following example creates an IPv4 extended ACL, defines rules in the ACL, and applies it as a receive-path ACL to an RBridge.

```
device(config)# ip access-list extended ipv4-receive-acl-example
device(conf-ipacl-ext)# hard-drop tcp host 10.0.0.1 any count
device(conf-ipacl-ext)# hard-drop udp any host 20.0.0.1 count
device(conf-ipacl-ext)# permit tcp host 10.0.0.2 any eq telnet count
device(conf-ipacl-ext)# permit tcp host 10.0.0.2 any eq bgp count

device(conf-ipacl-ext)# rb 1
device(config-rbridge-id-1)# ip receive access-group ipv4-receive-acl-example in
```

## History

Release version	Command history
6.0.1a	This command was introduced.

# ip route

Adds a static route to the IP routing tables.

## Syntax

```
ip route A.B.C.D / L A.B.C.D [ metric ] [ distance distance ] [ tag tag ]
```

```
ip route A.B.C.D / L { <N>gigabitethernet slot / port | port-channel number | ve vlan_id } [ metric ] [ distance distance ] [ tag tag ]
```

```
ip route A.B.C.D / L null 0 slot / port [ metric ] [ distance distance ] [ tag tag ]
```

```
no ip route A.B.C.D / L A.B.C.D
```

```
no ip route A.B.C.D / L { <N>gigabitethernet slot / port | port-channel number | ve vlan_id } [ metric ] [ distance distance ] [ tag tag ]
```

```
no ip route A.B.C.D / L null 0 slot / port
```

## Parameters

*A.B.C.D / L*

Specifies the destination IPv4 address and prefix-length.

*A.B.C.D*

Specifies the IPv4 address of the next hop.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand, **ten**, **forty**, or **hundred** (for example, **tengigabitethernet** specifies a 10-Gbps Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gbps Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel. The range is from 1 through 6144.

**ve** *vlan\_id*

Specifies the VLAN number.

**null 0**

Drops packets with this destination.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**metric**

Cost metric of the route. The range is from 1 through 16. The default is 1.

**distance** *distance*

Specifies the administrative distance of the route. When comparing otherwise equal routes to a destination, an Extreme device prefers lower administrative distances over higher ones. The range from 1 through 254. The default is 1.

**tag** *tag*

Tag value of the route to use for route filtering with a route map. The range is from 0 through 4294967295.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Enter **no ip route** followed by the route identifier to remove a static route.

## Examples

The following example configures a static route on the default vrf to 10.95.7.0, using 10.95.6.157 as the next-hop gateway.

```
device(config)# rbridge-id 30
device(config-rbridge-id-30)# ip route 10.95.7.0/24 10.95.6.157
```

The following example configures a static route on a non- default vrf to 10.95.7.0, using 10.95.6.157 as the next-hop gateway.

```
device(config)# rbridge-id 30
device(config-rbridge-id-30)# vrf red
device(config-vrf-red)# address-family ipv4 unicast
device(vrf-red-ipv4-unicast)# ip route 11.2.85.0/24 10.20.80.5
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

# ip route next-hop-vrf

Enables the leaking of static routes from one VRF instance to another.

## Syntax

```
ip route ip_addr/mask next-hop-vrf vrf VRF_name next_hop_ip_addr
```

```
ip route ip_addr/mask next-hop-vrf vrf VRF_name { <N> gigabitethernet rbridge-id/slot/port | port-channel number | ve vlan_id }
```

```
no route ip_addr/mask next-hop-vrf vrf VRF_name
```

## Command Default

Disabled

## Parameters

*ip\_addr/mask*

IPv4 address in dotted-decimal notation with a CIDR notation mask.

**vrf** *VRF\_name*

Specifies the name of the target VRF instance to which route leaking is enabled.

*ip\_addr*

Next-hop IP address in the target VRF instance.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (ten, forty, hundred; for example, **tengigabitethernet** specifies a 10-Gbps Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gbps Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel. The range is from 1 through 6144.

**ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

## Modes

RBridge ID configuration mode

VRF address-family IPv4 configuration mode

## Usage Guidelines

The next hop for the route leak can be either the IP address of the destination VRF or the interface specified by slot/port or VLAN ID.

Enter **no ip route** *ip\_addr mask next-hop-vrf* to disable the leaking of static routes.

## Examples

To enable static route leaking from the default VRF to VRF "blue":

```
switch# config
switch (config)# rbridge-id 2
switch (config-rbridge-id-2)# ip route 1.1.1.0/24 next-hop-vrf blue 10.1.1.10
```

To enable static route leaking from the default VRF to a specified VRF "blue" port:

```
switch# config
switch (config)# rbridge-id 2
switch (config-rbridge-id-2)# ip route 10.1.1.0/24 nexthop-vrf blue 2/1
```

This example shows the static route leaking enabled from the default VRF to VRF "blue":

```
switch# show running rbridge

rbridge-id 2
ip route 0.0.0.0/0 10.24.64.1
ip route 1.1.1.0/24 next-hop-vrf blue 10.1.1.10
```

# ip route static bfd

Configures Bidirectional Forwarding Detection (BFD) session parameters for IP static routes.

## Syntax

```
ip route static bfd dest-ip-address source-ip-address [ interval transmit-time min-rx receive-time multiplier number ]
no ip route static bfd dest-ip-address source-ip-address
```

## Command Default

BFD is not configured for an IP static route.

## Parameters

*dest-ip-address*

Specifies the destination IP address.

*source-ip-address*

Specifies the source IP address.

**interval** *transmit-time*

Specifies the interval, in milliseconds, a device waits to send a control packet to BFD peers. Valid values range from 50 through 30000.

**min-rx** *receive-time*

Specifies the interval, in milliseconds, a device waits to receive a control packet from BFD peers. Valid values range from 50 through 30000.

**multiplier** *number*

Specifies the number of consecutive BFD control packets that must be missed from a BFD peer before BFD determines that the connection to that peer is not operational. Valid values range from 3 through 50.

## Modes

RBridge ID configuration mode

Address-family IPv4 unicast VRF configuration mode

## Usage Guidelines

The **interval** *transmit-time* and **min-rx** *receive-time* variables are the intervals desired by the local device. The actual values in use will be the negotiated values.

For single-hop static BFD sessions, timeout values are optional because all required information is available from the outgoing interface. For multi-hop BFD sessions, if the configured **interval** and **min-rx** parameters conflict with those of an existing session, the lower values are used.

If you configure a neighbor IP address and a source IP address that already exist in BFD, BFD overwrites the existing interval values and multiplier for the IP addresses with the new values, on behalf of the static module.



Static BFD can be configured without configuring a static route to configure a BFD session. This is especially useful on BFD neighbors when they have reachability from other neighbors via OSPF or BGP. You must configure different BFD sessions for each ECMP path with the corresponding interface IP as the source IP address.

The **no** form of the command removes the configured BFD IP static route.

## Examples

The following example configures a BFD session on an IP static route.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip route static bfd 10.0.2.1 10.1.1.1 interval 500 min-rx 500
multiplier 5
```

The following example configures a BFD session on an IP static route in a non-default VRF instance.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# vrf orange
device(config-vrf-orange)# address-family ipv4 unicast
device(vrf-ipv4-unicast)# ip route static bfd 10.2.2.2 10.3.3.3 interval 600 min-rx 700 multiplier 10
```

## History

Release version	Command history
6.0.1	This command was introduced.

## Related Commands

[ip route static bfd holdover-interval](#), [show bfd](#), [show tunnel](#)

# ip route static bfd holdover-interval

Sets the time interval for which BFD session DOWN notifications are delayed before an IP static route is notified that a BFD session is down.

## Syntax

```
ip route static bfd holdover-interval time
no ip route static bfd holdover-interval time
```

## Command Default

The BFD holdover interval is set to 0 by default.

## Parameters

*time*

Specifies BFD holdover-time interval in seconds. Valid values range from 1 through 30. The default is 0.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no** form of the command to remove the configured BFD holdover interval from the configuration, and revert to the default value of 0.

Use the **ip route static bfd holdover-interval** command to set the time interval for which BFD session DOWN notifications are delayed before static routes are notified that a BFD session is down. If the BFD session is restored within the specified time interval, no DOWN notification is sent.

Use the **ip route static bfd holdover-interval** command in RBridge ID configuration mode to set the BFD holdover-time interval globally for static routes. Configured values apply to all VRFs.

## Examples

This example sets the BFD holdover interval globally for IP static routes to 15.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip route static bfd holdover-interval 15
```

This example removes the configured BFD holdover interval for IP static routes.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# no ip route static bfd holdover-interval
```

## History

Release version	Command history
6.0.1	This command was introduced.

## Related Commands

[ip route static bfd](#), [show bfd](#), [show tunnel](#)

# ip router-id

Changes the router ID that is already in configured.

## Syntax

```
ip router-id A.B.C.D
```

```
no ip router-id A.B.C.D
```

## Parameters

*A.B.C.D*

Specifies the IPv4 address that you want as the router ID.

## Modes

RBridge ID configuration mode

VRF configuration mode

## Usage Guidelines

Though a device has IP addresses assigned to various interfaces, some routing protocols identify the device by the router ID rather than the IP addresses assigned to the interfaces connected by the protocol.

The **no** form of the command removes the configured router ID and restores the default router ID.

## Examples

The following example specifies the router ID as 192.158.1.2.

```
device# configure terminal
device(config)# rbridge-id 30
device(config-rbridge-id-30)# ip router-id 192.158.1.2
```

# ip unnumbered

Designates the interface as an unnumbered IP interface and specifies a donor interface for the address.

## Syntax

```
ip unnumbered { <N> gigabitethernet rbridge-id/slot/port | loopback number | ve vlan_id }
no ip unnumbered
```

## Parameters

<N> *gigabitethernet*

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>*gigabitethernet* with the desired operand (for example, *tengigabitethernet* specifies a 10-Gb Ethernet port). The use of *gigabitethernet* without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**loopback** *number*

Specifies a loopback port number to be the donor interface. The range is from 1 through 255.

**ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no ip unnumbered** command to designate the interface as a standard IP interface.

An unnumbered IP interface borrows an IP address from another Layer 3 interface already configured on the device. This address is used as a source address in the Layer 3 packets transmitted from the unnumbered IP interface (referred to as the donor interface).

LLDP must be enabled on the unnumbered IP interface.

Only physical interfaces and Layer 3 port-channels can be configured as unnumbered IP interfaces. Unnumbered interfaces are not supported for Ve or SVI interfaces.

The donor interface can be any other Layer 3 interface; such as physical, Ve, loopback, and so on.

Unnumbered IP interface support is available for the IPv4 address-family. Only IPv4 addresses can be borrowed for an unnumbered IP interface.

Once the interface is configured as an unnumbered IP interface, it is treated as a point-to-point interface, so there can be only one remote neighbor attached to this interface.

Since there is no network subnet associated with the unnumbered IP interface, ARP is not supported on unnumbered IP interface.

Unnumbered IP interfaces are not supported in the management VRF.

IP address configurations are the only configurations that the unnumbered interfaces inherit from the donor interface. All other configurations (such as ICMP, ACLs, DHCP, and PBR) that are configured on the donor interface apply only to the donor interface and are not inherited by the unnumbered interfaces. You must configure these features separately on the unnumbered interfaces.

When eBGP is used in conjunction with the IP unnumbered interfaces feature, it is important to ensure that sufficient Time To Live (TTL) values are available for hello messages to establish separate neighbor adjacencies on leaf switches in an IP Fabric. To do so, use the **neighbor ebgp-multihop** command and set the number of maximum hops to **2**.

## Examples

The following example configures a port-channel interface and uses Loopback 1 as the donor interface.

```
device# configure terminal
device(config)# interface Port-channel 302
device(conf-Port-channel-302)# ip unnumbered loopback 1
```

The following example configures a FortyGigabitEthernet interface and uses Loopback 2 as the donor interface.

```
device# configure terminal
device(config)# interface FortyGigabitEthernet 236/0/13
device(conf-if-fo-236/0/13)# ip unnumbered loopback 2
```

The following example configures a FortyGigabitEthernet interface and uses Ve 100 as the donor interface.

```
device# configure terminal
device(config)# interface FortyGigabitEthernet 236/0/13
device(conf-if-fo-236/0/13)# ip unnumbered ve 100
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.0.1	This command was modified to include Usage Guidelines when used in conjunction with the <b>neighbor ebgp-multipath</b> command.

# ipv6 access-group

Applies rules specified in an IPv6 access control list (ACL) to traffic entering or exiting an interface.

## Syntax

```
ipv6 access-group ACLname { in | out } [ switched | routed ]
```

```
no ipv6 access-group ACLname { in | out } [ switched | routed ]
```

## Parameters

*ACLname*

Specifies the name of the standard or extended IPv6 access list.

**in | out**

Specifies the binding direction (ingress or egress).

**switched**

Filter only switched traffic. This parameter is not valid for management or overlay-gateway interfaces.

**routed**

Filter only routed traffic. This parameter is not valid for management or overlay-gateway interfaces.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to apply an IPv6 ACL to one of the following interface types:

- User interfaces
  - Physical Ethernet interfaces
  - Logical interfaces (LAGs)
  - Virtual Ethernet interfaces (VEs)
- All supported management interfaces
- Overlay gateways

You can apply a maximum of six ACLs to a user interface, as follows:

- One ingress MAC ACL—if the interface is in switchport or overlay-gateway mode
- One egress MAC ACL—if the interface is in switchport mode
- One ingress IPv4 ACL
- One egress IPv4 ACL
- One ingress IPv6 ACL
- One egress IPv6 ACL

You can apply a maximum of two ACLs to a management interface, as follows:

- One ingress IPv4 ACL

- One ingress IPv6 ACL

You can apply a maximum of three ACLs to an overlay gateway, as follows:

- One ingress MAC ACL
- One ingress IPv4 ACL
- One ingress IPv6 ACL

You can apply an ACL to multiple interfaces. And you can apply an ACL twice—ingress and egress—to a given user interface.

If you do not specify **switched** or **routed**, the ACL applies both to switched and routed traffic.

To remove an ACL from an interface, enter the **no** form of this command.

## Examples

The following example applies an IPv6 ACL named ip6\_acl\_7 on a specific 10-gigabit Ethernet interface to incoming traffic.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# ipv6 access-group ipv6_acl_7 in
```

The following example removes an IPv6 ACL named ipv6\_acl\_7 from a specific 10-gigabit Ethernet interface.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# no ipv6 access-group ipv6_acl_7 in
```

## Related Commands

[interface](#), [ipv6 access-list](#), [show running-config ipv6 access-list](#)



# ipv6 access-list

Creates a standard or extended IPv6 access control list (ACL). In ACLs, you can define rules that permit or deny network traffic based on criteria that you specify.

## Syntax

```
ipv6 access-list { standard | extended } ACLname
no ipv6 access-list { standard | extended } ACLname
```

## Parameters

### standard | extended

Specifies one of the following types of access lists:

#### standard

Contains rules that permit or deny traffic based on source addresses that you specify. The rules are applicable to all ports of the specified addresses.

#### extended

Contains rules that permit or deny traffic according to source and destination addresses, as well as other parameters. For example, you can also filter by port, protocol (TCP or UDP), and TCP flags.

### *ACLname*

Specifies an ACL name unique among all ACLs (Layer 2 and Layer 3). The name can be up to 63 characters in length, and must begin with an alphanumeric character. No special characters are allowed, except for the underscore and hyphen.

## Modes

Global configuration mode

## Usage Guidelines

An ACL name can be up to 63 characters long, and must begin with a-z, A-Z or 0-9. You can also use underscore (\_) or hyphen (-) in an ACL name, but not as the first character.

After you create an ACL, use the **seq** command to create filtering rules for that ACL.

An ACL starts functioning only after it is applied to an interface, using the **access-group** command.

To delete an ACL, use the **no access-list** command. You can delete an ACL only after you first remove it from all interfaces to which it is applied, using the **no access-group** command.

## Examples

The following example creates an IPv6 standard ACL:

```
device# configure
device(config)# ipv6 access-list standard stdV6ACL1
```

The following example creates an IPv6 extended ACL:

```
device# configure
device(config)# ipv6 access-list extended ipv6_acl_1
```

The following example creates rules on an IPv6 standard ACL:

```
device# configure
device(config)# ipv6 access-list standard stdV6ACL1
device(config-ipv6-std)# seq 10 permit 2001:db8:85a3:0:0:8a2e:370:7334
device(config-ipv6-std)# seq 11 deny any
```

The following example deletes an IPv6 ACL:

```
device# configure
device(config)# no ipv6 access-list standard stdV6ACL1
```

# ipv6 address

Configures a primary or secondary global or unique local IPv6 unicast address, including a manually configured interface ID.

## Syntax

```
ipv6 address ipv6-prefix/prefix-length [ secondary ]
```

```
no ipv6 address ipv6-prefix/prefix-length [ secondary ]
```

## Command Default

If the **secondary** keyword is not used, the address is a primary address.

## Parameters

*ipv6-prefix*

IPv6 prefix in hexadecimal with 16-bit values between colons, as specified in RFC 2373.

*prefix-length*

A decimal value specifying the length of the IPv6 prefix.

**secondary**

Specifies that the address is a secondary address. A maximum of 256 secondary addresses can be configured.

## Modes

Interface subtype configuration mode

## Usage Guidelines

A secondary address cannot be configured on an interface unless the primary address is configured first.

Use the **no** form of this command to remove the configuration.

The primary address cannot be deleted on an interface unless the secondary addresses are deleted first.

This command is not supported on loopback or management interfaces.

## Examples

To configure a global prefix and the interface ID on an Ethernet interface:

```
switch(config)# int te 3/1/1
switch(config-if-te-3/1/1)# ipv6 address 2001:db8:12d:1300:240z:d0ff:fe48:4672/64
```

To configure the above as a secondary address:

```
switch(config)# int te 3/1/1
switch(config-if-te-3/1/1)# ipv6 address 2001:db8:12d:1300:240z:d0ff:fe48:4672/64 secondary
```

# ipv6 address anycast

Configures an anycast address for a set of interfaces that belong to different nodes.

## Syntax

```
ipv6 address ipv6-prefix/prefix-length anycast
```

```
no ipv6 address ipv6-prefix/prefix-length anycast
```

## Parameters

*ipv6-prefix*

IPv6 prefix in hexadecimal with 16-bit values between colons, as specified in RFC 2373.

*prefix-length*

A decimal value specifying the length of the IPv6 prefix.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Sending a packet to an anycast address results in the delivery of the packet to the closest interface that has an anycast address. An anycast address looks similar to a unicast address, because it is allocated from the unicast address space. If you assign an IPv6 unicast address to multiple interfaces, it is an anycast address. On the device, you configure an interface assigned an anycast address to recognize the address as an anycast address.

If you assign an IPv6 unicast address to multiple interfaces, it is an anycast address. On the device, you configure an interface assigned an anycast address to recognize the address as an anycast address.

Use the **no** form of this command to remove the configuration.

### NOTE

IPv6 anycast addresses are described in detail in RFC 1884. See RFC 2461 for a description of how the Neighbor Discovery mechanism handles anycast addresses.

## Examples

To configure an anycast address on an Ethernet interface:

```
switch(config)# int te 3/1/1
switch(config-if-te-3/1/1)# ipv6 address 2001:db8:12d:1300:240z:d0ff:fe48:4672/64 anycast
```

# ipv6 address eui-64

Configures a global or unique local IPv6 unicast address with an automatically computed EUI-64 interface ID.

## Syntax

```
ipv6 address ipv6-prefix/prefix-length eui-64
```

```
no ipv6 address ipv6-prefix/prefix-length eui-64
```

## Parameters

*ipv6-prefix*

IPv6 prefix in hexadecimal with 16-bit values between colons, as specified in RFC 2373.

*prefix-length*

A decimal value specifying the length of the IPv6 prefix.

**eui-64**

Configures the global or unique local unicast address with a 64-bit Extended Unique Identifier, using the MAC address of the interface to construct the interface ID automatically.

## Modes

Interface subtype configuration mode

## Examples

To configure a global prefix and an automatically computed EUI-64 interface ID on a TenGigabitEthernet interface:

```
switch(config)# int te 3/1/1  
switch(config-if-te-3/1/1)# ipv6 address 2001:db8:12d:1300::/64 eui-64
```

# ipv6 address link-local

Configures an explicit link-local address on an interface.

## Syntax

```
ipv6 address ipv6-address link-local
```

```
no address ipv6-address link-local
```

## Parameters

*ipv6-address*

Explicit IPv6 address for the interface. Format can be xxxx.xxxx or xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.

## Modes

Interface subtype configuration mode

## Usage Guidelines

By default, when IPv6 is enabled, link-local addresses are computed automatically.

When configuring VLANs that share a common tagged interface with a virtual Ethernet (VE) interface, it is recommended that you override the automatically computed link-local address with a manually configured unique address for the interface. If the interface uses the automatically computed address, which in the case of VE interfaces is derived from a global MAC address, all VE interfaces will have the same MAC address.

## Examples

The following example configures a unique link-local address on an Ethernet interface.

```
switch(config)# int te 3/0/1
switch(config-if-te-3/0/1)# ipv6 address fe80::240:d0ff:fe48:4672 link-local
```

## Related Commands

[ipv6 address use-link-local-only](#)

# ipv6 address use-link-local-only

Enables IPv6 on an interface and configures an automatically computed link-local address.

## Syntax

```
ipv6 address use-link-local-only  
no ipv6 address use-link-local-only
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to disable IPv6 on an interface and consequently the automatic computing of a link-local address.

## Examples

To enable IPv6 on an interface and configure an automatically computed link-local address:

```
switch(config)# int te 3/1/1  
switch(config-if-te-3/1/1)# ipv6 address use-link-local-only
```

## Related Commands

[ipv6 address link-local](#)

# ipv6 anycast-address

Configures the IPv6 anycast address on an interface.

## Syntax

**ipv6 anycast-address** *ipv6-address / mask*

**no ipv6 anycast-address** *ipv6-address / mask*

## Command Default

No IPv6 anycast address is defined.

## Parameters

*ipv6-address / mask*

Specifies the IPv6 anycast address and mask. A mask value is required.

## Modes

Virtual ethernet (VE) configuration mode

## Usage Guidelines

This command is supported only if neighbor-discovery (ND) suppression is enabled on the VE.

To delete a configured IPv6 anycast address, use the **no** form of this command.

## Examples

The following example specifies an IPv6 anycast address on VE 10.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# interface ve 10
device(config-rbridge-ve-10)# ipv6 anycast-address 1234:10::10:100/64
```

## History

Release version	Command history
7.0.0	This command was introduced.



# ipv6 anycast-gateway-mac

Configures the IPv6 anycast-gateway MAC address.

## Syntax

```
ipv6 anycast-gateway-mac { default-mac | mac-address }  
no ipv6 anycast-gateway-mac { default-mac | mac-address }
```

## Command Default

No IPv6 anycast-gateway MAC address is defined.

## Parameters

### default-mac

Sets the IPv6 anycast-gateway MAC address to 02e0.5200.0200.

### mac-address

Specifies a non-default IPv6 anycast-gateway MAC address.

## Modes

RBridge-ID configuration mode

## Usage Guidelines

The first three bytes (six digits) of a non-default IPv6 anycast-gateway MAC address must be identical with the first three bytes of a corresponding IPv4 anycast-gateway MAC address. For example if the IPv4 anycast-gateway MAC address is 2222.2244.4444, the IPv6 address could be 2222.2266.6666.

To delete the configured IPv6 MAC address, use the **no** form of this command.

To change a configured MAC address, first delete the current address.

## Examples

The following example specifies a default IPv6 anycast-gateway MAC address.

```
device# configure terminal  
device(config)# rbridge-id 1  
device(config-rbridge-id-1)# ipv6 anycast-gateway-mac default-mac
```

The following example specifies a non-default IPv6 anycast-gateway MAC address.

```
device# configure terminal  
device(config)# rbridge-id 1  
device(config-rbridge-id-1)# ipv6 anycast-gateway-mac 2222.2266.6666
```

## History

Release version	Command history
7.0.0	This command was introduced.

# ipv6 dhcp relay address

Configures the IPv6 DHCP Relay address on a Layer 3 interface.

## Syntax

**ipv6 dhcp relay address** *ipv6-addr* [**interface** *interface-type interface-name*] [**use-vrf** *vrf-name* ]

**no ipv6 dhcp relay address** *ipv6-addr* [**interface** *interface-type interface-name*] [**use-vrf** *vrf-name* ]

## Parameters

*ipv6-addr*

IPv6 address of the DHCP server where the DHCP client requests are to be forwarded.

**interface**

This parameter specifies the outgoing interface, used when the relay address is a link-local or multicast address

*interface-type*

The type of interface, such as GigabitEthernet, TenGigabitEthernet, FortyGigabitEthernet, HundredGigabitEthernet, or VE.

*interface-name*

The interface number or VLAN ID.

**use-vrf**

Use this option if the VRF where the DHCP server is located is different from the VRF of the interface where the client is connected.

*vrf-name*

VRF name.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command uses the IPv6 address of the DHCP server where the DHCP client requests are to be forwarded. You can configure the address on a virtual Ethernet (VE) or a physical GigabitEthernet, TenGigabitEthernet, or FortyGigabitEthernet interface. You can configure up to 16 relay destination addresses on an interface.

Enter the command while in interface subtype configuration mode for a VE or Ethernet interface where you want to configure the IPv6 DHCP Relay. Use the **no** version of this command to remove the IPv6 DHCP Relay from the interface. If the **use-vrf** option is not used, it is assumed that the DHCP server and interface where the client is connected are on the same VRF.

If the relay address is a link local address or a multicast address, an outgoing interface must be configured for IPv6 relay to function. In instances where the server address is relayed to a different VRF compared to a client connected interface VRF, in addition to the relay address, you must also specify the user-vrf, otherwise IPv6 relay may not function correctly. IPv6 route leaking is also required for IPv6 reachability.

The **no** form of the command deletes the IPv6 DHCP Relay address from the interface.

## Examples

To configure an IPv6 DHCP Relay address on a TenGigabitEthernet interface:

```
device(config)# interface tengigabitethernet 2/3/1
device(config-if-te-2/3/1)# ipv6 dhcp relay address 2001::1122:AABB:CCDD:3344
```

To configure an IPv6 DHCP Relay address on an RBridge virtual Ethernet (VE) interface:

```
device(config)# rbridge-id 1
device(config-rbridge-id-1)# interface ve 101
device(config-rbridge-id-Ve-101)# ipv6 dhcp relay address fe80::224:38ff:febb:e3c0 interface ve 201
```

To configure an IPv6 DHCP Relay address on an RBridge VE interface if the DHCP server is on a different VRF than the interface where the client connects:

```
device# config
device(config)# rbridge-id 2
device(config-rbridge-id-2)# interface ve 103
device(config-rbridge-id-Ve-103)# ipv6 dhcp relay address fe80::224:38ff:febb:e3c0 use-vrf blue
```

### NOTE

For the relay configurations to be successful, you must specify an outgoing interface for any link local IPv6 relay addresses.

## History

Release version	Command history
5.0.1	This command was introduced.
6.0.1a	This command was modified to remove reference to "standalone" mode.

# ipv6 fabric-virtual-gateway

Enables IPv6 Fabric-Virtual-Gateway configurations.

## Syntax

```
ipv6 fabric-virtual-gateway~
```

```
no ipv6 fabric-virtual-gateway~
```

## Command Default

None

## Modes

Fabric-Virtual-Gateway global interface VE IPv6 configuration mode

## Usage Guidelines

Enter the **no** form of the command to disable IPv6 Fabric-Virtual-Gateway configurations.

## Examples

To enable IPv6 Fabric-Virtual-Gateway on a global virtual Ethernet (VE) interface:

```
device(config)# interface ve 4000
device(config-ve-4000)# ipv6 fabric-virtual-gateway
```

## History

Release version	Command history
5.0.1	This command was introduced.

# ipv6 icmpv6 echo-reply

Enables the generation of an IPv6 Internet Control Message Protocol version 6 (ICMPv6) Echo Reply message.

## Syntax

```
ipv6 icmpv6 echo-reply
no ipv6 icmpv6 echo-reply
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This is an interface-specific configuration that is persistent across a switch reload.  
 Use the **no** form of this command to disable Echo Reply messages.

## Examples

To enable IPv6 ICMPv6 Echo Reply messages on a FortyGigabitEthernet interface:

```
device(config)# interface fortygigabitethernet 1/0/50
device(conf-int-fo-1/0/50)# ipv6 icmpv6 echo-reply
```

## History

Release version	Command history
5.0.1	This command was introduced.

# ipv6 icmpv6 rate-limiting

Limits the rate at which IPv6 Internet Control Message Protocol version 6 (ICMPv6) messages are sent on an IPv6 network.

## Syntax

```
ipv6 icmpv6 rate-limiting milliseconds
no ipv6 icmpv6 rate-limiting
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Parameters

*milliseconds*  
Number of milliseconds between packets. Range is from 1 through 4294967295. The default value is 1000 milliseconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This is an interface-specific configuration that is persistent across a switch reload.

Use the **no** form of this command to disable rate limiting

## Examples

To enable IPv6 ICMPv6 rate limiting on a FortyGigabitEthernet interface:

```
device(config)# interface fortygigabitethernet 1/0/50
device(conf-if-te-12/2/1)# ipv6 icmpv6 rate-limiting 3000
```

## History

Release version	Command history
5.0.1	This command was introduced.

## Related Commands

[ip icmp rate-limiting](#)

# ipv6 icmpv6 redirect

Enables IPv6 Internet Control Message Protocol version 6 (ICMPv6) Redirect messages, which request that packets be sent on an alternative route.

## Syntax

```
ipv6 icmpv6 redirect milliseconds
```

```
no ipv6 icmpv6 redirect
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This is an interface-specific configuration that is persistent across a switch reload.

Use the **no** form of this command to disable IPv6 ICMPv6 Redirect messages.

## Examples

To enable IPv6 ICMPv6 Redirect messages on a FortyGigabitEthernet interface:

```
device(config)# interface fortygigabitethernet 1/0/50
device(conf-if-te-12/2/1)# ipv6 icmpv6 redirect
```

## History

Release version	Command history
5.0.1	This command was introduced.



# ipv6 icmpv6 unreachable

Prohibits routers from forwarding an IPv6 Internet Control Message Protocol version 6 (ICMPv6) Destination Unreachable Code 3 (port unreachable) message on a point-to-point link back onto the ingress port.

## Syntax

```
ipv6 icmpv6 unreachable
no ipv6 icmpv6 unreachable
```

## Command Default

This command is enabled on the management port, but is disabled on the front-end ports.

## Modes

Interface subtype configuration mode

## Usage Guidelines

ICMPv6 Destination Unreachable Code 3 messages are sent for a discarded IPv6 packet. Packets are trapped and a corresponding error message is returned if either the port, host, or network is unreachable.

This is an interface-specific configuration that is persistent across a switch reload.

Use the **no** form of this command to disable the sending of Destination Unreachable Code 3 messages.

## Examples

To enable IPv6 ICMPv6 Destination Unreachable Code 3 messages on a FortyGigabitEthernet interface:

```
device(config)# interface fortygigabitethernet 1/0/50
device(conf-int-fo-1/0/50)# ipv6 icmpv6 unreachable
```

## History

Release version	Command history
5.0.1	This command was introduced.

# ipv6 import routes (IPv6 VRF address-family configuration mode)

Leaks IPv6 routes from one VRF to the VRF you are configuring, based on match criteria defined in route-map.

## Syntax

```
ipv6 import routes VRF_name map rmap_name
no ipv6 import routes
```

## Parameters

*VRF\_name*  
Specifies the VRF instance from which to leak routes to the VRF you are configuring.

*rmap\_name*  
Specifies the map name to use for route-leaking match criteria.

## Modes

IPv6 VRF address-family configuration mode

## Usage Guidelines

Use the **no** form of the command to remove routes from being leaked to the VRF you are configuring.

## Examples

To leak IPv6 routes from a VRF named "red" to the VRF named "orange," based on match criteria from a route map named "import-map," with an example RBridge ID of 10:

```
switch# configure
switch(config)# rbridge-id 10
switch(config-rbridge-id-10)# vrf orange
switch(config-vrf-orange)# address-family ipv6 unicast
switch(config-ipv6-unicast)# ipv6 import routes red route-map import-map
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 import routes (RBridge ID configuration mode)

Leaks IPv6 routes from the specified VRF to the default VRF, based on match criteria defined in route-map.

## Syntax

```
ipv6 import routes VRF_name map rmap_name  
no ipv6 import routes
```

## Parameters

*VRF\_name*  
Specifies the VRF instance from which to leak routes to the default VRF.

*rmap\_name*  
Specifies the map name to use for route-leaking match criteria.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no** form of the command to remove routes from being leaked to the default VRF.

## Examples

To leak IPv6 routes from a VRF named **red** to the default VRF, based on match criteria from a route map named **import-map**:

```
switch# configure  
switch(config)# rbridge-id 54  
switch(config-rbridge-id-54)# ipv6 import routes red map import-map
```

# ipv6 mld last-member-query-count

Configures the IPv6 MLDv1 snooping last-member query count on a specific VLAN interface.

## Syntax

```
ipv6 mld last-member-query-count value  
no ipv6 mld last-member-query-count
```

## Parameters

*value*

The range is from 1 through 10. The default is 2.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The last-member query count is the number of times, separated by the last-member query-response interval, that an MLD query is sent in response to a host leave message from the last known active host on the subnet.

Use the **no** form of this command to restore the default.

## Examples

To change the IPv6 MLDv1 snooping last-member query count from the default on a VLAN interface:

```
switch(config)# int vlan 2000  
switch(config-Vlan-2000)# ipv6 mld last-member-query-count 3
```

## Related Commands

[ipv6 mld last-member-query-interval](#)

# ipv6 mld last-member-query-interval

Configures the IPv6 MLDv1 snooping last-member query interval on a specific VLAN interface.

## Syntax

```
ipv6 mld last-member-query-interval msec  
no ipv6 mld last-member-query-interval
```

## Parameters

*msec*

The range is from 100 through 2500 milliseconds. The default is 1000.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The last-member query interval is the interval for the response to a query sent after a host leave message is received from the last known active host on the subnet. The group is deleted if no reports are received in this interval. This interval adjusts the speed at which messages are transmitted on the subnet. Smaller values detect the loss of a group member faster.

Use the **no** form of this command to restore the default.

## Examples

To configure IPv6 MLDv1 snooping last-member query interval on a VLAN interface:

```
switch(config)# int vlan 2000  
switch(config-Vlan-2000)# ipv6 mld last-member-query-interval 25
```

## Related Commands

[ipv6 mld last-member-query-count](#)

# ipv6 mld query-interval

Configures the maximum interval for IPv6 MLDv1 snooping queries for a specific VLAN interface.

## Syntax

```
ipv6 mld query-interval sec  
no ipv6 mld query-interval
```

## Parameters

sec

The range is from 1 through 18000 seconds. The default is 125.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The value set by the **ipv6 mld query-interval** command must be greater than the value set by the **ipv6 mld query-max-response-time** command.

A larger value means that queries are sent less often.

Use the **no** form of this command to restore the default.

## Examples

To configure the maximum interval for IPv6 MLDv1 snooping queries on a VLAN interface:

```
switch(config)# int vlan 2000  
switch(config-vlan-2000)# ipv6 mld query-interval 1200
```

## Related Commands

[ipv6 mld query-max-response-time](#)

# ipv6 mld query-max-response-time

Configures the maximum response time for IPv6 MLDv1 snooping queries for a specific VLAN interface.

## Syntax

```
ipv6 mld query-max-response-time sec  
no ipv6 mld last-member-query-interval
```

## Parameters

`sec`  
The range is 1 through 25 seconds. The default is 10.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The maximum response delay is inserted into the periodic general query interval. This is useful when snooping querier functionality is enabled. Larger values spread out host responses over a longer time.

The value set by the **ipv6 mld query-max-response-time** command must be less than the value of the general query interval that is set by the **ipv6 mld query-interval** command.

Use the **no** form of this command to restore the default.

## Examples

To configure IPv6 MLDv1 query maximum response time on a VLAN interface:

```
switch(config)# int vlan 2000  
switch(config-Vlan-2000)# ipv6 mld query-max-response-time 15
```

## Related Commands

[ipv6 mld query-interval](#)

# ipv6 mld snooping enable

Enables IPv6 MLDv1 Layer 2 snooping globally or on a specific VLAN.

## Syntax

`ipv6 mld snooping enable`

`no ipv6 mld snooping enable`

## Command Default

IPv6 MLDv1 snooping is disabled.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

MLD snooping must be enabled globally first, after which it must be enabled on a VLAN

Use the **no** form of this command to disable IPv6 MLDv1 snooping globally or on a specific VLAN.

### NOTE

When MLD snooping is disabled globally, the snooping configuration remains in the running configuration and snooping is disabled on all VLANs.

## Examples

To enable IPv6 MLDv1 snooping globally:

```
switch(config)# ipv6 mld snooping enable
```

To enable IPv6 MLDv1 snooping on a specific VLAN:

```
switch(config)# int vlan 10
switch(config-vlan-10)# ipv6 mld snooping enable
```



# ipv6 mld snooping fast-leave

Configures the immediate-leave feature for the groups on a specific VLAN.

## Syntax

```
ipv6 mld snooping fast-leave  
no ipv6 mld snooping fast-leave
```

## Command Default

This feature is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command minimizes the leave latency of group memberships on an interface, as the device does not send group-specific queries. As a result, the group entry is removed from the multicast routing table as soon as a group leave message is received.

Use the **no** form of this command to restore the default.

## Examples

To configure the immediate-leave feature on a VLAN:

```
switch(config)# int vlan 2000  
switch(config-Vlan-2000)# ipv6 mld snooping fast-leave
```

# ipv6 mld snooping mrouter interface

Configures a VLAN port member to be a multicast router (mrouter) port.

## Syntax

`ipv6 mld snooping mrouter interface { <N>gigabitethernet | port-channel number }`

`no ipv6 mld snooping mrouter interface { <N>gigabitethernet | port-channel number }`

## Parameters

`<N>gigabitethernet`

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace `<N>gigabitethernet` with the desired operand (for example, `ten``gigabitethernet` specifies a 10-Gb Ethernet port). The use of `gigabitethernet` without a speed value specifies a 1-Gb Ethernet port.

`rbridge-id`

Specifies an RBridge ID.

`slot`

Specifies a valid slot number.

`port`

Specifies a valid port number.

`port-channel number`

Specifies the port-channel number. The number of available channels ranges from 1 through 6144.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the `no` form of this command to disable the VLAN port member from being an mrouter port.

## Examples

To configure a VLAN port member to be an mrouter port:

```
switch(config)# int vlan 2000
switch(config-Vlan-2000)# ipv6 mld snooping interface te 54/0/1
```

# ipv6 mld snooping querier enable

Activates or deactivates IPv6 MLDv1 Layer 2 multicast snooping querier functionality for a VLAN.

## Syntax

```
ipv6 mld snooping querier enable  
no ipv6 mld snooping querier enable
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to deactivate this functionality.

## Examples

To enable MLD snooping querier functionality on a VLAN:

```
switch(config)# int vlan 2000  
switch(config-Vlan-2000)# ipv6 mld snooping querier enable
```

# ipv6 mld snooping restrict-unknown-multicast

Deactivates or reactivates on a VLAN the flooding of unregistered multicast data traffic on IPv6 MLDv1 snooping-enabled VLANs.

## Syntax

```
ipv6 mld snooping restrict-unknown-multicast  
no ipv6 mld snooping restrict-unknown-multicast
```

## Command Default

This feature is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

MLD snooping must be enabled globally first, after which it must be enabled on a VLAN.

Use the **no** form of this command to reactivate the flooding of unregistered multicast data traffic on all IPv6 MLDv1 snooping-enabled VLANs.

### NOTE

When MLD snooping is disabled globally, the snooping configuration remains in the running configuration and snooping is disabled on all VLANs.

## Examples

To deactivate on a VLAN the flooding of unregistered multicast data traffic:

```
switch(config-Vlan-2000)# ipv6 mld snooping restrict-unknown-multicast
```

To reactivate on a VLAN the flooding of unregistered multicast data traffic :

```
switch(config-Vlan-2000)# no ipv6 mld snooping restrict-unknown-multicast
```

# ipv6 mld snooping robustness-variable

Configures a value to compensate for packet loss in congested networks.

## Syntax

`ipv6 mld snooping robustness-variable value`

`no ipv6 mld snooping robustness-variable value`

## Command Default

This feature is disabled.

## Parameters

*value*

The range is from 2 through 10. The default is 2.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This value determines the number of general MLD snooping queries that are sent before a multicast address is aged out for lack of a response.

Use the **no** form of this command to restore the default.

## Examples

To change the robustness variable from the default on a VLAN:

```
switch(config-Vlan-2000)# ipv6 mld snooping robustness-variable 7
```

# ipv6 mld startup-query-count

Configures the IPv6 MLDv1 number of queries that are separated by the startup query interval.

## Syntax

```
ipv6 mld startup-query-count value  
no ipv6 mld startup-query-count value
```

## Command Default

This feature is disabled.

## Parameters

*value*  
The range is from 1 through 10. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

## Examples

To change the startup query count on a VLAN:

```
switch(config)# interface vlan 2000  
switch(config-Vlan-2000)# ipv6 mld startup-query-count 5
```

## Related Commands

[ipv6 mld startup-query-interval](#)

# ipv6 mld startup-query-interval

Configures the IPv6 MLDv1 startup query interval.

## Syntax

```
ipv6 mld startup-query-interval value
```

```
no ipv6 mld startup-query-interval value
```

## Command Default

This feature is disabled.

## Parameters

*value*

The range is from 1 through 450. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

## Examples

To change the startup query interval on a VLAN:

```
switch(config)# interface vlan 2000
switch(config-Vlan-2000)# ipv6 mld startup-query-interval 4
```

## Related Commands

[ipv6 mld startup-query-count](#)

# ipv6 mld static-group interface

Configures IPv6 MLDv1 Layer 2 multicast static IPv6 groups on an interface for a VLAN.

## Syntax

**ipv6 mld static-group** *group-IPv6-address* **interface** *interface*

**no ipv6 mld static-group** *group-IPv6-address* **interface** *interface*

## Parameters

*group-IPv6-address*

A multicast address to be joined, in the format *xxxx:xxxx/ml*, *xxxx:xxxx::/ml*

*interface*

An Ethernet or port-channel interface.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the static-group configuration on an interface for a VLAN.

## Examples

To configure multicast static IPv6 groups on an Ethernet interface for a VLAN:

```
switch(config)# int vlan 2000
switch(config-vlan-2000)# ipv6 mld static-group ff1e::1 interface te 54/0/1
```



# ipv6 mtu

Configures a maximum size for IPv6 MTU packets to be sent on an interface.

## Syntax

```
ipv6 mtu bytes
```

```
no ipv6 mtu
```

## Command Default

MTU size is 1500 bytes.

## Parameters

*bytes*

IPv6 MTU in bytes. Range is from 1280 through 9000. The default is 1500.

## Modes

Global configuration mode

RBridge ID configuration mode

Interface subtype configuration mode

## Usage Guidelines

To route packets larger than 2500 bytes (the default for an Ethernet interface), you must also use the **mtu** command to set the same MTU value on the interface as that set by the **ipv6 mtu** command. Otherwise packets will be dropped. The range for the **mtu** command is from 1522 through 9219 bytes.

The **no ipv6 mtu** command returns the MTU size to the default value.

## Examples

The following example sets the MTU to 2000 bytes for every interface.

```
device# configure terminal
device(config)# ipv6 mtu 2000
```

To configure a maximum MTU size of 1800 on an Ethernet interface:

```
switch(config)# rbridge-id 3
switch(config-rbridge-id-3)# int te 3/0/1
switch(config-if-te-3/0/1)# ipv6 mtu 1800
```

# ipv6 nd cache expire

Configures the time interval after which the IPv6 Neighbor Discovery cache is deleted or refreshed.

## Syntax

```
ipv6 nd cache expire minutes
```

```
no ipv6 nd cache expire minutes
```

## Parameters

*minutes*

Interval in minutes. The range is from 1 through 240. The default is 240.

## Modes

RBridge ID configuration mode

Interface subtype configuration mode

## Usage Guidelines

Cache entries expire and are deleted if they remain in a "stale" state as defined by *minutes*.

Use the **no** version of this command to restore the default interval at which the cache is deleted.

## Examples

To set the Neighbor Discovery cache deletion or refresh interval to 180 minutes on an Ethernet interface:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd cache expire 180
```

# ipv6 nd dad attempts

Configures the number of IPv6 Neighbor Discovery Neighbor Solicitation (NS) messages to be sent as part of duplicate address detection (DAD).

## Syntax

```
ipv6 nd dad attempts number
```

```
no ipv6 nd dad attempts number
```

## Parameters

*number*

Number of solicitations. The range is from 0 through 10. The default is 2.

## Modes

RBridge ID configuration mode

Interface subtype configuration mode

## Usage Guidelines

To restore the number of neighbor solicitation messages to be sent to the default value, use the **no** form of this command.

## Examples

To set the number of Neighbor Discovery NS messages to be sent on an Ethernet interface to 5:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd dad attempts 5
```

To disable DAD on an Ethernet interface:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd dad attempts 0
```

# ipv6 nd dad time

Configures the retransmit time interval for IPv6 Neighbor Discovery Neighbor Solicitation (NS) messages that are sent as part of duplicate address detection (DAD).

## Syntax

```
ipv6 nd dad time seconds
```

```
no ipv6 nd dad time
```

## Parameters

*seconds*

Time in seconds. The range is from 1 through 5. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

To restore the default NS message retransmit interval for DAD, use the **no** form of this command.

Use the **ipv6 nd ns-interval** command to configure the interval for NS address resolution.

## Examples

To set the retransmit time interval globally for Neighbor Discovery NS messages to be sent on a specified RBridge to 3 seconds:

```
switch(config-rbridge-id-122)# ipv6 nd dad time 3
```

To set the retransmit time interval for Neighbor Discovery NS messages to be sent on an Ethernet interface to 3 seconds:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd dad time 3
```

## Related Commands

[ipv6 nd ns-interval](#)

# ipv6 nd hoplimit

Configures the number of hops to be advertised in IPv6 Neighbor Discovery Router Advertisement (RA) messages.

## Syntax

```
ipv6 nd hoplimit number  
no ipv6 nd hoplimit
```

## Parameters

*number*

The number of hops to be advertised. The range is from 0 through 255. The default is 64.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

## Examples

To set the number of hops to be advertised in Neighbor Discovery RA messages sent on an Ethernet interface to 32:

```
switch(config)# rbridge-id 54  
switch(config-rbridge-id-54)# int te 54/0/3  
switch(config-if-te-54/0/3)# ipv6 nd hoplimit 32
```

# ipv6 nd managed-config-flag

In IPv6 Neighbor Discovery, indicates to hosts on a local link that they must use the stateful autoconfiguration feature to obtain IPv6 addresses for their interfaces.

## Syntax

```
ipv6 nd managed-config-flag  
no ipv6 nd managed-config-flag
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

If the **ipv6 nd managed-config-flag** command is configured, hosts use stateful autoconfiguration to obtain IPv6 address information.

- If the **ipv6 nd managed-config-flag** command is configured, it overrides an existing configuration set by the **ipv6 nd other-config-flag** command.
- If the **ipv6 nd managed-config-flag** command is not configured, whether hosts can obtain IPv6 addresses by means of the stateful autoconfiguration feature is determined by means of the **ipv6 nd other-config-flag** command.

Use the **no** form of this command to remove the configuration.

## Examples

To indicate to hosts on a local link to an Ethernet interface that they must use the stateful autoconfiguration feature to obtain IPv6 addresses for their interfaces:

```
switch(config)# rbridge-id 54  
switch(config-rbridge-id-54)# int te 54/0/3  
switch(config-if-te-54/0/3)# ipv6 nd managed-config-flag
```

## Related Commands

[ipv6 nd other-config-flag](#)

# ipv6 nd mtu

Sets the size of the maximum transmission unit (MTU) that is advertised in Neighbor Discovery Router Advertisement (RA) messages.

## Syntax

```
ipv6 nd mtu number
```

```
no ipv6 nd mtu
```

## Parameters

*number*

Size, in bytes, of the MTU that is advertised. The range is from 1280 through 65535. The default is 1500.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

## Examples

To set the maximum IPv6 MTU packet size to 2400 bytes on an Ethernet interface:

```
switch(config)# rbridge-id 3
switch(config-rbridge-id-3)# int te 3/1/1
switch(config-if-te-3/1/1)# ipv6 nd mtu 2400
```

# ipv6 nd ns-interval

Sets the interval for address resolution between IPv6 Neighbor Discovery Neighbor Solicitation (NS) messages.

## Syntax

```
ipv6 nd ns-interval seconds
```

```
no ipv6 nd ns-interval
```

## Parameters

*seconds*

Number of seconds between neighbor solicitation messages. The range is from 1 through 5. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command sets the interval for address resolution only. Use the **ipv6 nd dad time** command to configure the retransmit time interval for NS messages that are sent as part of duplicate address detection (DAD).

Use the **no** form of this command to restore the default.

## Examples

To set the interval between Neighbor Discovery NS messages sent on an Ethernet interface to 3 seconds:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd ns-interval 3
```

## Related Commands

[ipv6 nd dad time](#)



# ipv6 nd other-config-flag

In IPv6 Neighbor Discovery, indicates to hosts on a local link that they can use the stateful autoconfiguration feature to obtain configuration settings other than IPv6 address information for their interfaces.

## Syntax

```
ipv6 nd other-config-flag
```

```
no ipv6 nd other-config-flag
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

If the **ipv6 nd managed-config-flag** command is configured, local hosts use stateful autoconfiguration to obtain IPv6 addresses for their interfaces.

- If the **ipv6 nd managed-config-flag** command is configured, it overrides an existing configuration set by the **ipv6 nd other-config-flag** command.
- If the **ipv6 nd managed-config-flag** command is not configured, whether hosts can obtain IPv6 address information by means of the stateful autoconfiguration feature is determined by means of the **ipv6 nd other-config-flag** command.

Use the **no** form of this command to remove the configuration.

## Examples

To indicate to local hosts on an Ethernet interface that they must use the stateful autoconfiguration feature to obtain configuration settings other than IPv6 address information for their interfaces:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd other-config-flag
```

## Related Commands

[ipv6 nd managed-config-flag](#)

# ipv6 nd prefix

Configures which IPv6 prefixes are included in IPv6 Neighbor Discovery Router Advertisement (RA) messages.

## Syntax

**ipv6 nd prefix** *ipv6-prefix/prefix-length*

**no ipv6 nd prefix** *ipv6-prefix/prefix-length*

## Parameters

*ipv6-prefix*

IPv6 prefix in hexadecimal with 16-bit values between colons, as specified in RFC 2373.

*prefix-length*

A decimal value specifying the length of the IPv6 prefix.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the IPv6 prefixes.

Valid and preferred lifetimes are default values, which are 2592000 and 604800, respectively.

## Examples

To include an IPv6 prefix in router advertisement messages sent out an Ethernet interface:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd prefix 2ffe:1111::/64
```

# ipv6 nd ra-interval

Configures the maximum interval range and minimum interval at which IPv6 Neighbor Discovery Router Advertisement (RA) messages are sent.

## Syntax

```
ipv6 nd ra-interval max-value min min-value
```

```
no ipv6 nd ra-interval
```

## Parameters

*max-value*

Maximum interval range in seconds. The range is from 4 through 1800. The default interval is 200 through 600, with messages sent randomly within that interval.

**min**

Specifies a minimum interval in seconds.

*min-value*

The range is from 0 through 1800. The default is 200.

## Modes

Interface subtype configuration mode

## Usage Guidelines

It is recommended that the interval set by this command be less than or equal to the device lifetime value set by the **ipv6 nd ra-lifetime** command if the device is advertised as a default device.

Use the **no** form of this command to restore the default.

## Examples

To set a maximum interval range and minimum interval for RA messages on an Ethernet interface:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd ra-interval 1200 min 400
```

## Related Commands

[ipv6 nd ra-lifetime](#)

# ipv6 nd ra-lifetime

Configures the amount of time in IPv6 Neighbor Discovery that a router is considered a valid default router.

## Syntax

```
ipv6 nd ra-lifetime seconds
```

```
no ipv6 nd ra-lifetime
```

## Parameters

*seconds*

Time in seconds. The range is from 0 through 9000. The default is 1800.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Note the following behavior:

- If the value set by this command is 0, the router is not advertised as a default router on the interface.
- If the value set by this command is not 0, the router is considered a default router on the interface.

It is recommended that the interval set by this command be greater than or equal to the value set by the **ipv6 nd ra-interval** command if the device is advertised as a default device.

Use the **no** form of this command to restore the default.

## Examples

To set the time that a router is considered a valid default router on an Ethernet interface:

```
switch(config)# rbridge-id 54
switch(config-rbridge-id-54)# int te 54/0/3
switch(config-if-te-54/0/3)# ipv6 nd ra-lifetime 2400
```

## Related Commands

[ipv6 nd ra-interval](#)

# ipv6 nd reachable-time

Configures the amount of time in IPv6 Neighbor Discovery that a device considers a removed IPv6 node reachable.

## Syntax

```
ipv6 nd reachable-time milli seconds  
no ipv6 nd reachable-time
```

## Parameters

*milliseconds*  
Time in milliseconds. The range is from 0 through 3600000. The default is 0.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Setting the reachable time to a nonzero value ensures that all nodes on the same link share the same value. The default of 0 means that no reachable time is specified.

Use the **no** form of this command to restore the default.

## Examples

To set the amount of time that a device considers a removed IPv6 node reachable on an Ethernet interface:

```
switch(config)# rbridge-id 54  
switch(config-rbridge-id-54)# int te 54/0/3  
switch(config-if-te-54/0/3)# ipv6 nd reachable-time 1800000
```

# ipv6 nd retrans-timer

Configures the time advertised between IPv6 Neighbor Discovery Neighbor Solicitation (NS) messages.

## Syntax

```
ipv6 nd retrans-timer milliseconds  
no ipv6 nd retrans-timer
```

## Parameters

*milliseconds*

Interval, in milliseconds, at which NS messages are sent. The range is from 0 through 4294967295. The default is 0.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

## Examples

To set the time advertised between NS messages on an interface:

```
switch(config)# rbridge-id 54  
switch(config-rbridge-id-54)# int te 54/0/3  
switch(config-if-te-54/0/3)# ipv6 nd retrans-timer 4500000
```

# ipv6 nd suppress-ra

Disables the sending of ICMPv6 Router Advertisement (RA) messages, including those sent in response to a solicitation as well as MTUs.

## Syntax

```
ipv6 nd suppress-ra [ all | mtu ]  
no ipv6 nd suppress-ra [ all | mtu ]
```

## Parameters

**all**  
Disables the sending of all RA messages, including those sent in response to a solicitation.

**mtu**  
Disables the sending of MTUs in RA messages.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to enable the sending of RA messages.

## Examples

To disable the sending of RA messages on an Ethernet interface, but allowing those sent in response to a solicitation:

```
switch(config)# rbridge-id 54  
switch(config-rbridge-id-54)# int te 54/0/3  
switch(config-if-te-54/0/3)# ipv6 nd suppress-ra
```

To disable the sending of RA messages, as well as those sent in response to a solicitation:

```
switch(config-if-te-54/0/3)# ipv6 nd suppress-ra all
```

To disable the sending of RA messages, allowing those sent in response to a solicitation, but also disabling the sending of MTUs:

```
switch(config-if-te-54/0/3)# ipv6 nd suppress-ra mtu
```

# ipv6 neighbor

Configures the IPv6 and MAC addresses of a neighbor as static entries for IPv6 Neighbor Discovery.

## Syntax

```
ipv6 neighbor ipv6address MACaddress
```

```
no ipv6 neighbor
```

## Parameters

*ipv6address*

IPv6 address of a neighbor in *A:B:C:D* format.

*MACaddress*

MAC address of the neighbor in *HHHH.HHHH.HHHH* format.

## Modes

RBridge ID configuration mode

Port-channel configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the IPv6 and MAC addresses.

## Examples

The following example configures an IPv6 and MAC address on an Ethernet interface for a neighbor.

```
device(config)# rbridge-id 54
device(config-rbridge-id-54)# ipv6 neighbor 2001:0db8:8086:6501::/32 abcd.abcd.abcd
```



# ipv6 ospf active

Sets a specific OSPFv3 interface to active.

## Syntax

```
ipv6 ospf active
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **ipv6 ospf active** command on each interface participating in adjacency formation. This command overrides the global passive setting on that interface, and enables transmission of OSPFv3 control packets.

## Examples

The following example sets a specific OSPFv3 virtual Ethernet (VE) interface to active.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 1
device(config-Ve-1)# ipv6 ospf active
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[ipv6 ospf passive](#)

# ipv6 ospf area

Enables OSPFv3 on an interface.

## Syntax

```
ipv6 ospf area area-id | ip-addr  
no ipv6 ospf area
```

## Command Default

OSPFv3 is disabled.

## Parameters

*area-id*  
Area ID in dotted decimal or decimal format.

*ip-addr*  
Area ID in IP address format.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command enables an OSPFv3 area on the interface to which you are connected.

The **no** form of the command disables OSPFv3 on this interface.

## Examples

The following example enables a configured OSPFv3 area named 0 on a specific OSPFv3 10-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface tengigabitethernet 1/0/49  
device(config-if-te-1/0/49)# ipv6 ospf area 0
```

The following example enables a configured OSPFv3 area named 0 on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal  
device(config)# rbridge-id 177  
device(config-rbridge-id-177)# interface ve 12  
device(config-Ve-12)# ipv6 ospf area 0
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf authentication ipsec

Specifies IP security (IPsec) as the authentication type for an OSPFv3 interface.

## Syntax

```
ipv6 ospf authentication ipsec key-add-remove-interval interval
no ipv6 ospf authentication ipsec key-add-remove-interval interval
```

## Command Default

Disabled.

## Parameters

**key-add-remove-interval** *interval*  
 Specifies the OSPFv3 authentication key add-remove interval. Valid values range from decimal numbers 0 through 14400. The default is 300.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command removes IPsec authentication from the interface.

## Examples

The following example enables IPsec on a specified OSPFv3 10-gigabit interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/1
device(config-if-te-1/0/1)# ipv6 ospf area 0
device(config-if-te-1/0/1)# ipv6 ospf authentication ipsec
```

The following example sets the OSPFv3 authentication key add-remove interval to 480.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/1
device(config-if-te-1/0/1)# ipv6 ospf area 0
device(config-if-te-1/0/1)# ipv6 ospf authentication ipsec key-add-remove-interval 480
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# ipv6 ospf authentication ipsec disable

Disables IP security (IPsec) services on an OSPFv3 interface.

## Syntax

```
ipv6 ospf authentication ipsec disable
no ipv6 ospf authentication ipsec disable
```

## Command Default

Authentication is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to disable IPsec if it is enabled on the interface. Packets that are sent out will not be IPsec encapsulated and the received packets which are IPsec encapsulated will be dropped.

The **no** form of the command re-enables IPsec on the interface if IPsec is already configured on the interface.

## Examples

The following example disables IPsec on a specific OSPFv3 interface where IPsec is already enabled.

```
device# configure terminal
device(config)# interface tengigabitethernet 190/0/49
device(conf-if-te-190/0/49)# ipv6 ospf authentication ipsec disable
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# ipv6 ospf authentication spi

Specifies the security policy index (SPI) value for an OSPFv3 interface.

## Syntax

```
ipv6 ospf authentication spi spi { ah | esp null } { hmac-md5 | hmac-sha1 } key [ no-encrypt ] key
no ipv6 ospf authentication spi
```

## Command Default

Disabled.

The 40-hexadecimal character key is encrypted by default. Use the **no-encrypt** parameter to disable encryption.

## Parameters

*spi*

SPI value. Valid values range from decimal numbers 512 through 4294967295.

**ah**

Specifies Authentication Header (ah) as the protocol to provide packet-level security.

**esp**

Specifies Encapsulating Security Payload (ESP) as the protocol to provide packet-level security.

**null**

Specifies that the ESP payload is not encrypted.

**hmac-md5**

Enables Hashed Message Authentication Code (HMAC) Message Digest 5 (MD5) authentication on the OSPFv3 interface.

**hmac-sha1**

Enables HMAC Secure Hash Algorithm 1 (SHA-1) authentication on the OSPFv3 interface.

**key**

Number used in the calculation of the message digest. The 40-hexadecimal character key is stored in encrypted format by default.

**no-encrypt**

The 40-character key is not encrypted upon either its entry or its display.

*key*

The 40 hexadecimal character key.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter `no ipv6 ospf authentication spi spi` to remove the SPI value from the interface.

## Examples

The following example enables ESP and HMAC-SHA-1 on a specified OSPFv3 10-gigabit interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/1
device(config-if-te-1/0/1)# ipv6 ospf area 0
device(config-if-te-1/0/1)# ipv6 ospf authentication spi 512 esp null hmac-sha1 key
abcef12345678901234fedcba098765432109876
```

The following example enables HA and HMAC-MD5 on a specified OSPFv3 10-gigabit interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/1
device(config-if-te-1/0/1)# ipv6 ospf area 0
device(config-if-te-1/0/1)# ipv6 ospf authentication spi 750 ha hmac-md5 key
abcef12345678901234fedcba098765432109876
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# ipv6 ospf bfd

Enables Bidirectional Forwarding Detection (BFD) on a specific OSPFv3 interface.

## Syntax

```
ipv6 ospf bfd
```

```
no ipv6 ospf bfd
```

## Command Default

BFD is disabled by default.

## Modes

Interface subtype configuration mode

## Usage Guidelines

BFD sessions are initiated only if BFD is also enabled globally using the `bfd` command in OSPFv3 router configuration mode. If BFD is disabled using the `no bfd` command in OSPFv3 router configuration mode, BFD sessions on specific interfaces are deregistered.

The `no` form of the command removes all BFD sessions from a specified interface.

## Examples

The following example enables BFD on an OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/11
device(config-if-fo-101/0/11)# ipv6 ospf bfd
```

The following example disables BFD on an OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 24
device(config-Ve-24)# no ipv6 ospf bfd
```

## History

Release version	Command history
6.0.1	This command was introduced.



# ipv6 ospf cost

Configures cost for a specific OSPFv3 interface.

## Syntax

```
ipv6 ospf cost value  
no ipv6 ospf cost
```

## Command Default

Cost value is 1.

## Parameters

*value*

Cost value. Valid values range from 1 through 65535. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the OSPFv3 cost on the interface. If the cost is not configured with this command, OSPFv3 calculates the value from the reference and interface bandwidths.

For more information, refer to the **auto-cost reference-bandwidth** command.

The **no** form of the command disables the configured cost.

## Examples

The following example sets the cost to 550 on a specific OSPFv3 10-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface tengigabitethernet 190/0/49  
device(config-if-te-190/0/49)# ipv6 ospf cost 550
```

The following example sets the cost to 620 on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal  
device(config)# rbridge-id 177  
device(config-rbridge-id-177)# interface ve 14  
device(config-Ve-14)# ipv6 ospf cost 620
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf dead-interval

Specifies the time period for which a neighbor router waits for a hello packet from the device before declaring the router down.

## Syntax

```
ipv6 ospf dead-interval interval
no ipv6 ospf dead-interval
```

## Command Default

The specified time period is 40 seconds.

## Parameters

*interval*

Dead interval in seconds. Valid values range from 3 through 65535 seconds. The default is 40.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the dead interval, the hello interval is automatically changed to a value that is one fourth that of the new dead interval, unless the hello interval is also explicitly configured using the **ipv6 ospf hello-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is  $\frac{1}{4}$  times that of the dead interval.

The **running-config** command displays only explicitly configured values of the hello interval, which means that a value that was automatically changed as the result of a dead-interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the dead interval to 80 on a specific OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal
device# device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface fortygigabitethernet 101/0/10
device(config-if-fo-101/0/10)# ipv6 ospf dead-interval 80
```

The following example sets the dead interval to 80 on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 24
device(config-Ve-24)# ipv6 ospf dead-interval 80
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf hello-interval

Sets the length of time between the transmission of hello packets that an interface sends to neighbor routers.

## Syntax

```
ipv6 ospf hello-interval interval  
no ipv6 ospf hello-interval
```

## Command Default

The length of time between the transmission of hello packets is set to 10 seconds.

## Parameters

*interval*

Hello interval in seconds. Valid values range from 1 through 65535 seconds. The default is 10.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the hello interval, the dead interval is automatically changed to a value that is four times that of the new hello interval, unless the dead interval is also explicitly configured using the **ipv6 ospf dead-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is  $\frac{1}{4}$  times that of the dead interval.

The **running-config** command displays only explicitly configured values of the dead interval, which means that a value that was automatically changed as the result of a hello interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the hello interval to 20 on a specific OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface fortygigabitethernet 1/0/10  
device(conf-if-fo-1/0/10)# ipv6 ospf hello-interval 20
```

The following example sets the hello interval to 20 on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 24
device(config-Ve-24)# ipv6 ospf hello-interval 20
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf hello-jitter

Sets the allowed jitter between HELLO packets.

## Syntax

```
ipv6 ospf hello-jitter interval
no ipv6 ospf hello-jitter
```

## Command Default

10%

## Parameters

*jitter*

Allowed interval between hello packets. Valid values range from 1 through 50 percent (%).

## Modes

Interface subtype configuration mode

## Usage Guidelines

The hello interval can vary from the configured hello-interval to a maximum of percentage value of configured jitter.

## Examples

The following example sets the hello jitter to 20 on a specific OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/20
device(conf-if-fo-101/0/10)# ipv6 ospf hello-jitter 20
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf instance

Specifies the number of OSPFv3 instances running on an interface.

## Syntax

```
ipv6 ospf instance instanceID
no ipv6 ospf instance
```

## Parameters

*instanceID*  
Instance identification number. Valid values range from 0 through 255.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

## Examples

The following example sets the number of IPv6 OSPF instances to 35 on a specific OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/20
device(conf-if-fo-101/0/10)# ipv6 ospf instance 35
```

## History

Release version	Command history
5.0.0	This command was introduced.



# ipv6 ospf mtu-ignore

Enables or disables maximum transmission unit (MTU) match checking.

## Syntax

```
ipv6 ospf mtu-ignore
no ipv6 ospf mtu-ignore
```

## Command Default

Enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

In default operation, the IP MTU on both sides of an OSPFv3 link must be the same, and a check of the MTU is performed when Hello packets are first exchanged.

The **no** form of the command disables MTU-match checking on a specific interface.

## Examples

The following example disables MTU-match checking on a specific OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 101/0/10
device(conf-if-fo-101/0/10)# no ipv6 ospf mtu-ignore
```

The following example enables MTU-match checking on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 177
device(config-rbridge-id-177)# interface ve 24
device(config-Ve-24)# ipv6 ospf mtu-ignore
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf network

Configures network type.

## Syntax

```
ipv6 ospf network { broadcast | point-to-point }
no ipv6 ospf network
```

## Command Default

Network type is broadcast.

## Parameters

### broadcast

Network type is broadcast, such as Ethernet.

### point-to-point

Network type is point-to-point.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Point-to-point can support unnumbered links, which requires less processing by OSPFv3.

The **no** form of the command removes the network-type configuration.

### NOTE

The network type non-broadcast is not supported at this time.

## Examples

The following example configures an OSPFv3 point-to-point link on an OSPFv3 10-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/49
device(conf-if-te-1/0/49)# ipv6 ospf network point-to-point
```

The following example configures an OSPFv3 broadcast link on an OSPFv3 virtual Ethernet (VE) interface 20.

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 20
device(config-Ve-20)# ipv6 ospf network broadcast
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf passive

Sets a specific OSPFv3 interface to passive.

## Syntax

```
ipv6 ospf passive
no ipv6 ospf passive
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **ipv6 ospf passive** command disables transmission of OSPFv3 control packets on that interface. OSPFv3 control packets received on a passive interface are discarded.

The **no** form of the command sets an interface back to active.

## Examples

The following example sets a specific OSPFv3 virtual Ethernet (VE) interface to passive.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 20
device(config-Ve-20)# ipv6 ospf passive
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[ipv6 ospf active](#)

# ipv6 ospf priority

Configures priority for designated router (DR) election and backup designated routers (BDRs) on the interface you are connected to.

## Syntax

```
ipv6 ospf priority value
```

```
no ipv6 ospf priority
```

## Command Default

The value is set to 1.

## Parameters

*value*

Priority value. Valid values range from 0 through 255. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The OSPFv3 router assigned the highest priority becomes the designated router, and the OSPFv3 router with the second-highest priority becomes the backup router.

The **no** form of the command restores the default value.

## Examples

The following example sets a priority of 4 for the OSPFv3 router that is connected to an OSPFv3 10-gigabit Ethernet interface 1/0/49.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/49
device(conf-if-te-1/0/49)# ipv6 ospf priority 4
```

The following example sets a priority of 4 for the OSPFv3 router that is connected to an OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# interface ve 27
device(config-Ve-27)# ipv6 ospf priority 4
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf retransmit-interval

Configures the retransmit interval. The retransmit interval is the time between Link-State Advertisement (LSA) retransmissions to adjacent routers for a given interface.

## Syntax

```
ipv6 ospf retransmit-interval interval
```

```
no ipv6 ospf retransmit-interval
```

## Command Default

The interval is 5 seconds.

## Parameters

*interval*

Retransmit interval in seconds. Valid values range from 0 through 3600 seconds. The default is 5.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command resets the retransmit interval to its default.

## Examples

The following example sets the retransmit interval to 8 for all OSPFv3 devices on an OSPF 10-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/49
device(config-if-te-1/0/49)# ipv6 ospf retransmit-interval 8
```

The following example sets the retransmit interval to 26 for all OSPFv3 devices on an OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 178
device(config-rbridge-id-178)# interface ve 22
device(config-Ve-22)# ipv6 ospf retransmit-interval 26
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 ospf suppress-linklsa

Suppresses link LSA advertisements.

## Syntax

`ipv6 ospf suppress-linklsa`

`no ipv6 ospf suppress-linklsa`

## Modes

Interface subtype configuration mode

## Usage Guidelines

The `no` form of the command restores the defaults where link LSA advertisements are not suppressed.

## Examples

The following example suppresses link LSAs from being advertised on devices on a specific OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/0/49
device(conf-if-te-1/0/49)# ipv6 ospf suppress-linklsa
```

## History

Release version	Command history
5.0.0	This command was introduced.



# ipv6 ospf transmit-delay

Configures transmit delay for link-update packets. The transmit delay is the estimated time required for OSPFv3 to send link-state update packets on the interface to which you are connected.

## Syntax

```
ipv6 ospf transmit-delay value
```

```
no ipv6 ospf transmit-delay
```

## Command Default

The transmit delay is set to 1 second.

## Parameters

*value*

Transmit delay in seconds. Valid values range from 0 through 3600 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

## Examples

The following example sets a transmit delay of 25 seconds for devices on a specific OSPFv3 40-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/0/49
device(config-if-te-1/0/49)# ipv6 ospf transmit-delay 25
```

The following set a transmit delay of 45 seconds for routers on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# rbridge-id 177
device(config-rbridge-id-177)# interface ve 22
device(config-Ve-22)# ipv6 ospf transmit-delay 43
```

## History

Release version	Command history
5.0.0	This command was introduced.

# ipv6 prefix-list

Configures IPv6 prefix lists for use in basic traffic filtering.

## Syntax

```
ipv6 prefix-list name { [ deny ipv6-prefix/prefix-length | permit ipv6-prefix/prefix-length ] ge ge-value [ le le-value ] | seq
sequence-number }
```

```
no ipv6 prefix-list name
```

## Parameters

*name*

Permitted values are between 1 and 32 characters. Although the first character must be alphabetic, the others can be alphanumeric, underscores (\_) or minus signs (-).

**deny** *ip-prefix/prefix-length*

Denies a packet that contains a route specified in the prefix list. The prefix list matches only on the specified prefix/prefix length, unless you use the **ge** *ge-value* or **le** *le-value* parameters.

**permit** *ip-prefix/prefix-length*

Permits a packet that contains a route specified in the prefix list. The prefix list matches only on the specified prefix/prefix length, unless you use the **ge** *ge-value* or **le** *le-value* parameters.

**ge** *ge-value*

If you specify only **ge** *ge-value*, then the range is from *ge-value* to 128.

**le-value** *le-value*

If you specify only **le** *le-value*, then the range is from *le-value* to the *prefix-length* parameter.

**seq** *sequence-number*

Specifies an IPv6 prefix list sequence number. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with prefix list entry 5. The Extreme device interprets the prefix list entries in numerical order, beginning with the lowest sequence number.

## Modes

RBridge ID configuration mode

## Usage Guidelines

An IPv6 prefix list is composed of one or more conditional statements that execute a permit or deny action if a packet matches a specified prefix. In prefix lists with multiple statements, you can specify a sequence number for each statement. The specified sequence number determines the order in which the statement appears in the prefix.

You can configure an IPv6 prefix list on a global (RBridge ID) basis, then use it as input to other commands or processes, such as route aggregation, route redistribution, route distribution, route maps, and so on. When an Extreme device interface sends or receives an IPv6 packet, it applies the statements within the IPv6 prefix list in their order of appearance to the packet. As soon as a match occurs, the device takes the specified action (permit or deny the packet) and stops further comparison for that packet.

You can use permit statements in the prefix list to specify the traffic that you want to send to the other feature. If you use deny statements, the traffic specified by the deny statements is not supplied to the other feature. You can configure up to one hundred IPv6 prefix lists.

Use the **no ipv6 prefix-list** *name* command to delete a prefix list.

You must specify the `ipv6-prefix` parameter in hexadecimal using 16-bit values between colons as documented in RFC 4291. You must specify the `prefix-length` parameter as a decimal value. A slash mark (/) must follow the `ipv6-prefix` parameter and precede the `prefix-length` parameter.

The *ge-value* or *le-value* you specify must meet the following condition for *prefix-length*:

```
ge-value <= le-value <= 128
```

If you do not specify **ge** *ge-value* or **le** *le-value*, the prefix list matches only on the exact prefix you specify with the *ipv6-prefix/prefix-length* parameter.

## Examples

This example permits all routes for prefix 2001::/16:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ipv6 prefix-list routesfor2001 permit 2001::/16
device(config-rbridge-id-1)#
```

## Related Commands

[ipv6 address](#), [distribute-list route-map](#)

# ipv6 protocol vrrp

Globally enables IPv6 VRRPv3.

## Syntax

`ipv6 protocol vrrp`

`no ipv6 protocol vrrp`

## Command Default

IPv6 VRRPv3 is not enabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no** form of this command globally disables VRRPv3.

## Examples

To enable IPv6 VRRPv3 globally:

```
device# configure
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 protocol vrrp
```

## Related Commands

[ipv6 protocol vrrp-extended](#)

# ipv6 protocol vrrp-extended

Globally enables IPv6 VRRP-Ev3.

## Syntax

```
ipv6 protocol vrrp-extended
```

```
no ipv6 protocol vrrp-extended
```

## Command Default

IPv6 VRRP-Ev3 is disabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no** form of this command globally disables IPv6 VRRP-Ev3.

## Examples

To enable IPv6 VRRP-Ev3 globally:

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 protocol vrrp-extended
```

## Related Commands

[ipv6 protocol vrrp](#)

# ipv6 rguard

Router protocols are susceptible to rogue Router advertisements (RAs) generated by unauthorized or improperly configured devices. RA Guard, enabled by this command, prevents RAs generated by such devices from entering a Layer 2 network.

## Syntax

```
ipv6 rguard  
no ipv6 rguard
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

RA Guard is available only under IPv6.

You can apply RA Guard both to physical interfaces and to port-channel groups.

RA Guard is effective only on an environment where all traffic traverses the controlled Layer 2 networking devices.

The **no** form of this command disables this feature.

## Examples

The following example enables RA Guard on a specified 10-gigabit Ethernet interface.

```
device# configure terminal  
device(config)# interface tengigabitethernet 2/2/1  
device(conf-if-te-2/2/1)# ipv6 rguard
```

## Related Commands

[show interface](#), [show ipv6 rguard](#)

# ipv6 receive access-group

Applies an IPv6 access control list (ACL) at RBridge-level. Such *receive-path* ACLs filter incoming route-processor and multicast traffic according to rules that you create, but do not filter unicast data-path traffic.

## Syntax

```
ipv6 receive access-group acl-name in
```

```
no ipv6 receive access-group acl-name in
```

## Command Default

No receive-path ACLs are applied to the RBridge.

## Parameters

*acl-name*

Specifies the name of the standard or extended IP access list.

**in**

Specifies ingress traffic.

## Modes

RBridge ID configuration mode

## Usage Guidelines

For both interface ACLs and receive-path ACLs, you use identical commands to create the ACLs. You also use identical commands to define permit/deny rules in the ACLs. The only variance is the command you use to apply the ACL:

- To apply an interface ACL, from an interface-subtype configuration mode you use the { **ip** | **ipv6** | **mac** } **access-group** command.
- To apply a receive-path ACL, from RBridge ID configuration mode you use the { **ip** | **ipv6** } **receive access-group** command.

You can apply a receive-path ACL to multiple RBridges.

You can apply a maximum of two receive-path ACLs to an RBridge, as follows:

- One IPv4 receive-path ACL
- One IPv6 receive-path ACL

To remove a receive-path ACL from an RBridge, enter the **no** form of this command.

## Examples

The following example creates an IPv6 extended ACL, defines rules in the ACL, and applies it as a receive-path ACL to an RBridge.

```
device(config)# ipv6 access-list extended ipv6-receive-acl-example
device(conf-ipacl-ext)# hard-drop tcp host 10::1 any count
device(conf-ipacl-ext)# hard-drop udp any host 20::1 count
device(conf-ipacl-ext)# permit tcp host 10::2 any eq telnet count
device(conf-ipacl-ext)# permit tcp host 10::2 any eq bgp count

device(conf-ipacl-ext)# rb 1
device(config-rbridge-id-1)# ipv6 receive access-group ipv6-receive-acl-example in
```

## History

Release version	Command history
6.0.1a	This command was introduced.



## ipv6 route

Configures a static IPv6 route for an interface, with a destination network, a next-hop gateway, and optional cost metric and administrative distance.

### Syntax

```

ipv6 route dest-ipv6-prefix / prefix-length { next-hop-ipv6-address | link-local-next-hop-ipv6-address } [ metric ] [ distance
number ] [ tag tag-value ]

ipv6 route dest-ipv6-prefix / prefix-length { next-hop-ipv6-address | link-local-next-hop-ipv6-address } [ <N> gigabitethernet
slot / port | null 0 | port-channel number | ve vlan_id ] [ metric ] [ distance number ] [ tag tag-value ]

ipv6 route dest-ipv6-prefix / prefix-length { next-hop-ipv6-address | link-local-next-hop-ipv6-address } [ management 1 ]
[ metric ] [ distance number ] [ tag tag-value ]

ipv6 route ipv6-prefix / prefix-length next-hop-vrf vrf_name { next-hop-ipv6-address | <N>gigabitethernet slot / port | null 0 |
port-channel number | ve vlan_id | management 1 }

no ipv6 route dest-ipv6-prefix / prefix-length { next-hop-ipv6-address | link-local-next-hop-ipv6-address } [ metric ]
[ distance number ] [ tag tag-value ]

no ipv6 route dest-ipv6-prefix / prefix-length { next-hop-ipv6-address | link-local-next-hop-ipv6-address } [ <N>
gigabitethernet slot / port | null 0 | port-channel number | ve vlan_id ] [ metric ] [ distance number ] [ tag tag-value ]

no ipv6 route dest-ipv6-prefix / prefix-length { next-hop-ipv6-address | link-local-next-hop-ipv6-address } [ management 1 ]
[ metric ] [ distance number ] [ tag tag-value ]

no ipv6 route ipv6-prefix / prefix-length next-hop-vrf vrf_name { next-hop-ipv6-address | <N>gigabitethernet slot / port | null
0 | port-channel number | ve vlan_id | management 1 }

```

### Parameters

*dest-ipv6-prefix*

Destination IPv6 prefix in hexadecimal with 16-bit values between colons, as specified in RFC 2373.

*prefix-length*

A decimal value specifying the length of the IPv6 prefix.

*next-hop-ipv6-address*

IPv6 address of the next-hop gateway.

*link-local-next-hop-ipv6-address*

IPv6 address of the link-local next-hop gateway.

**next-hop-vrf** *vrf\_name* *next-hop-ipv6-address*

Specifies a VRF instance and a next-hop IPv6 address.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (ten, forty, hundred; for example, **ten****gigabitethernet** specifies a 10-Gbps Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gbps Ethernet port.

*slot*  
Specifies a valid slot number.

*port*  
Specifies a valid port number.

**null 0**  
Causes packets to the selected destination to be dropped by shunting them to the "null0" interface.

**port-channel** *number*  
Specifies a port-channel. The range is from 1 through 6144.

**ve** *vlan\_id*  
Specifies the VLAN ID of a virtual Ethernet (VE) interface.

**management 1**  
Specifies management interface.

*metric*  
Specifies a value that the Layer 3 switch uses to compare this route to other static routes in the IPv6 static route table that have the same destination. The metric applies only to routes that the Layer 3 switch has number already placed in the IPv6 static route table. Two or more routes to the same destination with the same metric will load share (as in ECMP load sharing). The range is from 1 through 16. The default is 1.

**distance**  
Specifies an administrative distance. This is a value that the Layer 3 switch uses to compare this route with routes from other route sources that have the same destination. By default, static routes take precedence over routes learned by routing protocols. To choose a dynamic route over a static route, configure the static route with a higher administrative distance than the dynamic route.

*number*  
The range is from 1 through 254. The default is 1.

**tag** *tag-value*  
Specifies a tag value for the route. The route tag can be used for route redistribution to routing protocols by means of route maps (as in IPv4 static route redistribution). The range is from 0 through 4294967295. The default is 0.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use this command to configure a static IPv6 route for an interface, with a destination network, a next-hop gateway, and an optional administrative distance.

## Examples

The following example configures a static IPv6 route to a destination network with the prefix 2001:db8::0/32, a next-hop gateway with the global address 2001:db8:0:ee44::1, and an administrative distance of 110.

```
device(config)# rbridge-id 54
device(rbridge-id-54)# ipv6 route 2001:db8::0/32 2001:db:0:ee44::1 distance 110
```

**NOTE**

See the *Extreme Network OS Layer 3 Routing Configuration Guide* for additional examples and details.

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

# ipv6 route static bfd

Configures Bidirectional Forwarding Detection (BFD) session parameters for IPv6 static routes.

## Syntax

```
ipv6 route static bfd dest-ipv6-address source-ipv6-address [ interface-type interface-name ] [ interval transmit-time min-rx  
receive-time multiplier number ]
```

```
no ipv6 route static bfd dest-ipv6-address source-ipv6-address
```

## Command Default

BFD is not configured for an IPv6 static route.

## Parameters

*dest-ipv6-address*

Specifies the IPv6 address of BFD neighbor.

*source-ipv6-address*

Specifies the source IPv6 address.

*interface-type*

The type of interface, such as gigabitEthernet, TengigabitEthernet, FortygigabitEthernet, HundredgigabitEthernet, or Ve interface.

*interface-name*

The interface number or VLAN ID.

**interval**

Specifies the interval a device waits to send a control packet to BFD peers.

*transmit-time*

Value in milliseconds. Valid values range from 50 to 30000.

**min-rx**

Specifies the interval a device waits to receive a control packet from BFD peers.

*receive-time*

Value in milliseconds. Valid values range from 50 to 30000.

**multiplier**

Specifies the number of consecutive BFD control packets that must be missed from a BFD peer before BFD determines that the connection to that peer is not operational. .

*number*

Value in decimals. Valid values range from 3 to 50.

## Modes

RBridge ID configuration mode

Address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the configured BFD IPv6 static route.

The *transmit-time* and *receive-time* variables are the intervals desired by the local device. The actual values in use will be the negotiated values.

For single-hop static BFD sessions, timeout values are optional since all required information is available from the outgoing interface. For multi-hop BFD sessions, if the configured **interval** and **min-rx** parameters conflict with those of an existing session, the lower values are used.

If you configure a neighbor IPv6 address and a source IPv6 address that already exist in BFD, BFD overwrites the existing timer-value and multiplier for the IPv6 addresses with the new values, on behalf of the static module.

Static BFD can be configured without configuring a static route to configure a BFD session. This is especially useful on BFD neighbors when they have reachability from other neighbors via OSPF or BGP. You must configure different BFD sessions for each ECMP path with the corresponding interface IP as the source IPv6 address.

For IPv6 static BFD sessions, if the BFD neighbor is link-local, the source IPv6 address must also be link-local.

If an IPv6 BFD session is running for a link-local BFD neighbor, the *interface-type interface-name* parameters are mandatory since the link-local address can be same on multiple interfaces.

## Examples

This example configures a BFD session on an IPv6 static route, specifying a VE interface.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 route static bfd fe80::a fe80::b ve 20 interval 100 min-rx 100
multiplier 10
```

This example configures a BFD session on an IPv6 static route in a non-default VRF instance.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# vrf orange
device(config-vrf-orange)# address-family ipv6 unicast
device(vrf-ipv6-unicast)# ipv6 route static bfd fe70::a fe60::b ve 10 interval 1000 min-rx 2000
multiplier 20
```

## History

Release version	Command history
6.0.1	This command was introduced.

## Related Commands

[ipv6 route static bfd holdover-interval](#), [show bfd](#), [show tunnel](#)

# ipv6 route static bfd holdover-interval

Sets the time interval for which BFD session DOWN notifications are delayed before an IPv6 static route is notified that a BFD session is down.

## Syntax

```
ipv6 route static bfd holdover-interval time
no ipv6 route static bfd holdover-interval time
```

## Command Default

The BFD holdover interval is set to 0 by default.

## Parameters

*time*

Specifies BFD holdover-time interval in seconds. Valid values range from 1 through 30. The default is 0.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no** form of the command to remove the configured BFD holdover interval from the configuration, and revert to the default value of 0.

Use the **ipv6 route static bfd holdover-interval** command to set the time interval for which BFD session DOWN notifications are delayed before static routes are notified that a BFD session is down. If the BFD session is restored within the specified time interval, no DOWN notification is sent.

Use the **ipv6 route static bfd holdover-interval** command in RBridge ID configuration mode to set the BFD holdover-time interval globally for static routes. Configured values apply to all VRFs.

## Examples

This example sets the BFD holdover interval globally for IPv6 static routes to 25.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 route static bfd holdover-interval 25
```

This example removes the configured BFD holdover interval for IPv6 static routes.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# no ipv6 route static bfd holdover-interval
```

## History

Release version	Command history
6.0.1	This command was introduced.

## Related Commands

[ipv6 route static bfd](#), [show bfd](#), [show tunnel](#)

# ipv6 router ospf

Enables and configures the Open Shortest Path First version 3 (OSPFv3) routing protocol.

## Syntax

```
ipv6 router ospf [ vrf name ]
no ipv6 router ospf
```

## Command Default

Disabled.

## Parameters

**vrf name**  
Specifies a nondefault VRF.

## Modes

RBridge ID configuration mode

## Usage Guidelines

If you save the configuration to the startup-config file after disabling OSPFv3, all OSPFv3 configuration information is removed from the startup-config file.

Use this command to enable the OSPFv3 routing protocol and enter OSPFv3 router or OSPFv3 router VRF configuration mode. OSPFv3 maintains multiple instances of the routing protocol to exchange route information among various VRF instances.

The **no** form of the command deletes all current OSPFv3 configurations and blocks any further OSPFv3 configuration.

## Examples

The following example enables OSPFv3 on a default VRF and enters OSPFv3 router configuration mode.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)#
```

## History

Release version	Command history
5.0.0	This command was introduced.



# ipv6 unreachable

Prohibits routers from forwarding an Internet Control Message Protocol version 6 (ICMPv6) Destination Unreachable Code 3 (port unreachable) message on a point-to-point link back onto the ingress port.

## Syntax

```
ipv6 unreachable  
no ipv6 unreachable
```

## Command Default

This command is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

By default, ICMPv6 Destination Unreachable Code 3 messages are sent for a discarded IP packet. You can disable the sending of these messages. Use the **no ipv6 unreachable** command to disable the sending of these messages.

This is an interface-specific configuration. The configuration is persistent across a switch reload.

## Related Commands

[ip icmp unreachable](#)

# ipv6 vrrp-extended-group

Configures an IPv6 VRRP-Ev3 group and enters into the VRRP-E configuration mode.

## Syntax

```
ipv6 vrrp-extended-group group-ID
```

```
no ipv6 vrrp-extended-group group-ID
```

## Parameters

*group-ID*

A number from 1 through 128 that you assign to the VRRP-Ev3 group.

## Modes

Virtual Ethernet (VE) interface configuration mode

## Usage Guidelines

Enter **no ipv6 vrrp-extended-group *group-ID*** to remove the specific IPv6 VRRP-Ev3 group. If you remove a group, you cannot retrieve it. You would have to redo the configuration procedure.

This configuration is for virtual Ethernet (VE) interfaces only. IPv6 VRRP-Ev3 must be enabled on the device before the IPv6 VRRP-E group is configured.

## Examples

The following example shows how to assign the VE interface with a VLAN number of 2019 to the VRRP-Ev3 group with the ID of 19.

```
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 protocol vrrp-extended
device(config-rbridge-id-122)# interface ve 2019
device(config-Ve-2019)# ipv6 address 2001:2019:8192::122/64
device(config-Ve-2019)# ipv6 vrrp-extended-group 19
device(config-vrrp-extended-group-19)#
```

## Related Commands

[ipv6 protocol vrrp-extended](#)

# ipv6 vrrp-group

Configures an IPv6 VRRPv3 group and enters into the virtual router configuration mode.

## Syntax

```
ipv6 vrrp-group group-ID
```

```
no ipv6 vrrp-group group-ID
```

## Parameters

*group-ID*

A value from 1 through 255 that you assign to the VRRPv3 group.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no ipv6 vrrp-group *group-ID*** to remove a specific IPv6 VRRPv3 group. If you remove a group, you cannot retrieve it. You would have to redo the configuration procedure.

IPv6 VRRPv3 must be enabled on the device before the IPv6 VRRP group is configured.

## Examples

The following example shows how to assign an Ethernet interface to the VRRPv3 group with the ID of 18.

```
device(config)# rbridge-id 125
device(config-rbridge-id-125)# ipv6 protocol vrrp
device(config-rbridge-id-125)# interface tengigabitethernet 101/1/6
device(config-if-te-101/1/6)# ipv6 address 2001:2019:8192::125/64
device(config-if-te-101/1/6)# ipv6 vrrp-group 18
device(config-vrrp-group-18)#
```

## Related Commands

[ipv6 protocol vrrp](#)

# ipv6 vrrp-suppress-interface-ra

Suppresses interface router advertisement (RA) when VRRPv3 is configured on an interface.

## Syntax

```
ipv6 vrrp-suppress-interface-ra  
no ipv6 vrrp-suppress-interface-ra
```

## Command Default

Interface RA is enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no ipv6 vrrp-suppress-interface-ra** to remove the suppression of interface RA.

Router advertisements are sent by the VRRP master device and contain the link-local virtual IP address and the virtual MAC address. For network security reasons, if you do not want the MAC addresses of interfaces to be viewed, you can disable RA messages.

## Examples

This example suppresses interface RA on a virtual Ethernet (VE) interface:

```
device(config)# rbridge-id 122  
device(config-rbridge-id-122)# ipv6 protocol vrrp  
device(config-rbridge-id-122)# interface ve 2019  
device(config-Ve-2019)# ipv6 vrrp-suppress-interface-ra
```

# iscsi-priority

Sets the iSCSI priority bitmap for use in the DCBX iSCSI TLV.

## Syntax

**iscsi-priority** *value*

**no iscsi-priority**

## Command Default

Priority bitmap value is 4.

## Parameters

*value*

The priority bitmap value. Valid values range from 0 through 7.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

Enter **no iscsi-priority** to return to the default value.

# isl-r\_rdy

Sets the flow control primitive used to prevent frame drop to ISL R\_RDY mode.

## Syntax

```
isl-r_rdy  
no isl-r_rdy
```

## Command Default

no **isl-r\_rdy** or VC\_RDY flow control mode.

## Modes

Interface Fibre Channel configuration mode

## Usage Guidelines

ISL R\_RDY mode can be enabled only for a port configured for long distance operation with long distance modes LE, LD, or LS.

In ISL R\_RDY mode, the port sends the R\_RDY primitive signal that the port is ready to receive frames. The port sends an exchange link parameter (ELP) with flow control mode 02. If a port is ISL R\_RDY enabled, it can only receive an ELP with flow control mode 02. A received ELP with flow control mode 01 will segment the fabric.

Extreme recommends disabling ISL R\_RDY.

This command can be used only on Network OS platforms with Fibre Channel ports.

A Fibre Channel port configured as a trunk port cannot have the ISL R\_RDY flow control enabled.

Enter **no isl-r\_rdy** to disable ISL R\_RDY mode on a port, and instead establish VC\_RDY flow control.

## Examples

To enable ISL R\_RDY mode on a Fibre Channel port:

```
switch(config)# interface fibrechannel 7/0/2  
switch(conf-if-fi-7/0/2)# isl-r_rdy
```

To disable ISL R\_RDY mode on a Fibre Channel port:

```
switch(config)# interface fibrechannel 7/0/2  
switch(conf-if-fi-7/0/2)# no isl-r_rdy
```

# iterations

For an implementation of an event-handler profile, specifies the number of times an event-handler action is run, when triggered.

## Syntax

**iterations** *num-iterations*

**no iterations**

## Command Default

When the trigger condition occurs, the event-handler actions runs once.

## Parameters

*num-iterations*

Specifies the number of times an event-handler action is run, when triggered. Valid values are any positive integer.

## Modes

Event-handler activation mode

## Usage Guidelines

The **no** form of this command resets the **iterations** setting to the default 1 iteration.

## Examples

The following example specifies 5 iterations.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# iterations 5
```

The following example resets **iterations** to the default value of 1 iteration.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# no iterations
```

## History

Release version	Command history
6.0.1	This command was introduced.

# keep-alive timeout (fabric-map)

Enables or disables the keep-alive timeout.

## Syntax

**keep-alive timeout**

**no keep-alive timeout**

## Modes

FCoE fabric-map configuration mode

## Usage Guidelines

You must be in the feature configuration mode for FCoE fabric-map for this command to function.

Enter **no keep-alive timeout** to disable the keep-alive timeout.

## Related Commands

[fabric-map, fcoe](#)



# key-add-remove-interval

Alters the timing of the authentication key add-remove interval.

## Syntax

**key-add-remove-interval** *interval*

**no key-add-remove-interval** *interval*

## Parameters

*interval*

Specifies the add-remove interval in seconds. Valid values range from 0 through 14400. The default is 300 seconds.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command resets the add-remove interval to the default value of 300 seconds.

## Examples

The following example sets the key add-remove interval to 240 seconds.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# key-add-remove-interval 240
```

The following example sets the key add-remove interval to 210 seconds in a nondefault VRF instance:

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf vrf red
device(config-ipv6-router-ospf-vrf-red)# key-add-remove-interval 240
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# key-rollover-interval

Alters the timing of the existing configuration changeover.

## Syntax

**key-rollover-interval** *interval*

**no key-rollover-interval** *interval*

## Parameters

*interval*

Specifies the key-rollover-interval in seconds. Valid values range from 0 through 14400. The default is 300 seconds.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

In order to have consistent security parameters, rekeying should be done on all nodes at the same time. Use the **key-rollover-interval** command to facilitate this. The key rollover timer waits for a specified period of time before switching to the new set of keys. Use this command to ensure that all the nodes switch to the new set of keys at the same time.

The **no** form of the command resets the rollover interval to the default value of 300 seconds.

## Examples

The following example sets the key rollover interval to 420 seconds.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# key-rollover-interval 420
```

The following example re-sets the key rollover interval to the default value.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# no key-rollover-interval 420
```

The following example re-sets the key rollover interval to the default value in a nondefault VRF instance.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf vrf red
device(config-ipv6-router-ospf-vrf-red)# no key-rollover-interval 420
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# keypair

Associates the generated RSA/ECDSA/DSA key pair with a trust point for security protocol exchanges for applications.

## Syntax

Associates the generated RSA/ECDSA/DSA key pair with the trust point.

**keypair** *key\_label*

**no keypair**

## Parameters

*key\_label*

Specifies the name of the key pair to associate with the trust point.

## Modes

Trust point configuration mode

## Usage Guidelines

Use the **no keypair** command to remove the key pair association.

## Examples

Typical command usage:

```
device(config-rbridge-id-1)# crypto ca trustpoint t1
device(config-ca-t1)# keypair k1
device(config-ca-t1)# do show running-config rbridge-id crypto
rbridge-id 1
  crypto key label k1 rsa modulus 2048
  crypto ca trustpoint t1
    keypair k1
!
device# show crypto ca trustpoint
rbridge-id:1
trustpoint: t1; key-pair: k1
```

## History

Release version	Command history
6.0.0	This command was introduced.

# I2traceroute

This command sends a simple traceroute from the source MAC address to the destination MAC address.

## Syntax

**I2traceroute**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command does not support command-line parameters. You are prompted for the required information after you enter the **I2traceroute** command.

This command sends a plain Layer 2 traceroute, hop by hop, from the switch that learned the source MAC address to the switch that learned the destination MAC address. The IP parameters included in the **I2traceroute** command allow for generating frames with similar properties as the ones generated from a connected device, thus traversing the same path through the fabric.

Configuration results depend on the configuration parameters specified. The following fields display when you enter the **I2traceroute** command:

- Source MAC address—Enter the source MAC address. The MAC address must be a valid MAC address that exists in the mac-address-table.
- Destination MAC address—Enter the destination MAC address. The MAC address must be a valid MAC address that exists in the mac-address-table.
- Vlan—Enter the VLAN number. On the Extreme VDX family of switches, VLANs are treated as interfaces from a configuration point of view. By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). Valid VLAN IDs are as follows:
  - On Extreme VDX 8770 switches: 1 through 4086 for 802.1Q VLANs (VLAN IDs 4087 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.
  - On all other Extreme VDX switches: 1 through 3962 for 802.1Q VLANs (VLAN IDs 3963 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.
- Edge rbridge-id—Enter the edge RBridge ID on which the **I2traceroute** command is to run.
- Extended commands—Enter **Y** to enable extended commands, which include protocol type (IP or FCoE), source IP address, destination IP address, IP protocol type (TCP or UDP), source port number, and destination port number.

Based on the input for Extended commands, if you enter **Y**, the parameters branch as follows:

- Protocol Type [IP]—Enter the protocol type. You must select the IP including.
  - Source IP address—Enter the source IP address.
  - Destination IP address—Enter the destination IP address.
- IP Protocol Type [ TCP | UDP 0-255 ]—Enter the IP protocol type including:
  - TCP (Transmission Control Protocol) is a connection-oriented protocol, which means that it requires handshaking to set up end-to-end communications.

- UDP (User Datagram Protocol) is a message-based connectionless protocol. Communication occurs by transmitting information in one direction, from the source to the destination, without verifying the readiness of the receiver.
- 0-255 is the numeric protocol value. to use as filter.
- The source port number. The valid port range is 0 through 65535. This is an optional field.
- The destination port number. The valid port range is 0 through 65535. This is an optional field.

## Examples

This example shows extended commands, IP protocol type, and TCP as the IP protocol type.

```
switch# l2traceroute

Source mac address           : 0050.564f.549f
Destination mac address     : 0005.1ea0.8dd8
Vlan [1-3583]               : 1
Edge rbridge-id [1-239]    : 1
Extended commands [Y/N]?   : Y
Protocol Type [IP/FCoE]    : IP
Source IP address           : 192.85.1.2
Destination IP address      : 192.0.2.2
IP Protocol Type [TCP/UDP/0-255] : TCP
Source port number [0-65535] : 58
Dest port number [0-65535]  : 67
switch# l2traceroute

Source mac address           : 0000.0000.1111
Destination mac address     : 0000.0000.2222
Vlan [1-3583]               : 1
Edge rbridge-id [1-239]    : 50
Extended commands [Y/N]?   : n
Rbridge      Ingress          Egress          Rtt (usec)
-----
50           Te 50/0/15            Te 50/0/38 (isl) 0
40           Te 40/0/38 (isl)         Te 40/0/2 (isl) 60322
10           Te 10/0/2 (isl)          Te 10/0/4 (isl) 1274
20           Te 20/0/4 (isl)          Te 20/0/10 (isl) 1119
30           Te 30/0/10 (isl)         Te 30/0/19      1787
```

# lACP default-up

Activates an LACP link in the absence of PDUs.

## Syntax

**lACP default-up**

**no lACP default-up**

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command forces the port to activate an LACP link if there are no PDUs available on the interface port.

This command is supported on all physical interfaces.

This command is visible only if the interface is a dynamic and standard member of a port-channel.

This command is not supported on static LAGs.

This command is not supported on static or dynamic trunks.

This command is not supported on any other types of interfaces, such as port-channel or VLAN.

Enter **no lACP default-up** to disable this feature.

## Examples

To activate an LACP link in the absence of PDUs on an Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/9
device(conf-if-te-1/0/9)# lACP default-up
```

# lACP port-priority

Configures the Link Aggregation Control Protocol (LACP) port priority of a member port of a port-channel.

## Syntax

**lACP port-priority** *value*

**no lACP port-priority**

## Command Default

The default value is 32768.

## Parameters

*value*

Specifies the priority. Valid values range from 1 through 65535. A lower number takes priority over a higher number.

## Modes

Interface subtype configuration mode.

## Usage Guidelines

An LACP port priority is configured on each port using LACP. The port priority determines which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

A link with higher priority (smaller in value) gets preference over a link with lower priority (greater in value).

Enter **no lACP port-priority** to return to the default value.

## Examples

To set the LACP port priority to 1000 for a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# lACP port-priority 1000
```

## Related Commands

[interface](#)



# lACP system-priority

Sets the Link Aggregation Control Protocol (LACP) system priority. The LACP priority determines which system is responsible for resolving conflicts in the choice of aggregation groups.

## Syntax

`lACP system-priority value`

`no lACP system-priority`

## Command Default

The default value is 32768.

## Parameters

*value*

Specifies the value of the LACP system priority. Valid values range from 1 through 65535.

## Modes

Global configuration mode

## Usage Guidelines

Lower numerical values have higher priorities.

Enter `no lACP system-priority` to reset the system priority to the default value.

## Examples

The following example sets the LACP system priority value to 68.

```
switch(config)# lACP system-priority 68
```

# lacp timeout

Sets the timeout value used by the Link Aggregation Control Protocol (LACP) to exchange packets on an interface before invalidating a received data unit (DU).

## Syntax

```
lacp timeout { long | short }
```

```
no lacp timeout
```

## Command Default

For Network OS trunks, the default value is the **short** timeout.

For standard LAGs, the default value is the **long** timeout.

## Parameters

### long

Specifies that a long-timeout value of 30 seconds will be used. With this value, the port waits three times this long (90 seconds) before invalidating the information received earlier on this PDU.

### short

Specifies that a short-timeout value of one second will be used. With this value, the port waits three times this long (three seconds) before invalidating the information received earlier on this PDU.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set the timeout value based on how frequently you think the device will receive LACP PDUs from the partner device.

Enter **no lacp timeout** to return to the default values.

## Examples

The following example sets the LACP long-timeout value on a specific 10-gigabit Ethernet interface:

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# lacp timeout long
```

## Related Commands

[interface](#)

# ldap-server host

Configures an LDAP-server host.

## Syntax

```
ldap-server host { ipaddr | FQDN } [ port portnum ] [ domain basedn ] [ timeout secs ] [ retries num ] [ use-vrf vrf-name ]
no ldap-server host { ipaddr | FQDN } [ use-vrf vrf-name ]
```

## Command Default

- Timeout: 5 seconds
- Port: 389
- Retries: 5

## Parameters

*ipaddr* | *FQDN*

Specifies the IPv4 address or Fully Qualified Domain name of the Active Directory (AD) server. IPv6 is supported for Windows 2008 AD server only. The maximum supported length for the LDAP host name is 40 characters.

port *portnum*

Specifies the TCP port used to connect the AD server for authentication. The port range is from 1024 through 65535.

domain *basedn*

Describes the base domain name of the host.

timeout *secs*

Specifies the wait time for a server to respond. The range is 1 through 60 seconds.

retries *num*

Specifies the number of retries for the server connection. The range is 0 through 100.

use-vrf *vrf-name*

Specifies a VRF through which to communicate with the LDAP server. See the Usage Guidelines.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to sets up a connection to the Lightweight Directory Access Protocol (LDAP) server host, or modifies an existing configuration. A maximum of 5 LDAP servers can be configured on a switch. Executing "no" on an attribute sets it with its default value.

Enter **no ldap-server host** to delete the server configuration.

Enter **no ldap-server host** with a parameter to restore the default value for that parameter.

Invoking **no** on an attribute sets the attribute with its default value.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

To add an LDAP server on port 3890 with retries set to three:

```
switch(config)# ldap-server host 10.24.65.6 domain sec.extremenetworks.com port 3890 retries 3
```

To change the domain in an existing configuration:

```
switch(config)# ldap-server host 10.24.65.6
switch(config-host-10.24.65.6)# domain security.extremenetworks.com
```

To delete an LDAP server:

```
switch(config)# no ldap-server host 10.24.65.6
```

To reset the number of retries to the default value:

```
switch(config)# ldap-server host 10.24.65.6 retries
```

Executing **no** on an attribute sets it with its default value.

```
switch(config)# no ldap-server host 10.24.65.6 retries
```

Attributes holding default values will not be displayed.

```
switch# show running-config ldap-server host 10.24.65.6

ldap-server host 10.24.65.6
  port      3890
  domain    security.extremenetworks.com
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.

## Related Commands

[certutil import ldapca](#), [ldap-server maprole](#), [show running-config ldap-server](#)

# ldap-server maprole

Maps an Active Directory (AD) group to a switch role.

## Syntax

```
ldap-server maprole group group_name role role_name  
no ldap-server maprole group group_name
```

## Parameters

**group** *group\_name*  
The name of the AD group.

**role** *role\_name*  
The name of the switch role.

## Modes

Global configuration mode

## Usage Guidelines

Enter `no ldap-server maprole group group_name` without the `role role_name` parameter to remove the mapping of the AD group to a role.

## Examples

To map the AD group "Administrator" to the switch role "admin":

```
switch(config)# ldap-server maprole group Administrator role admin
```

To remove the mapping:

```
switch(config)# no ldap-server maprole group Administrator
```

## Related Commands

[certutil import ldapca](#), [ldap-server host](#), [show running-config ldap-server](#)

# license add

Adds a license key to a switch.

## Syntax

```
license add { licstr licenseString | FTP-URL ftpPath | SCP-URL scpPath } [ rbridge-id rbridge-id ]
```

## Command Default

This command is executed on the local switch.

## Parameters

**licstr** *licenseString*

Specifies the license string to be added to the switch. The license string must be enclosed in double quotation marks. A maximum of 256 characters is allowed.

**FTP-URL** *ftpPath*

Specifies a URL from which to transfer license information using FTP. *ftp://username:password@hostname filepath*

**SCP-URL** *scpPath*

Specifies a URL from which to transfer license information using SCP. *scp://username:password@hostname/ filepath*

**rbridge-id** *rbridge-id*

Executes the command on the remote switch specified by the RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Depending on the feature being added, you may need to disable and re-enable the affected ports for this command to take effect. Follow the instructions in the command output.

If you install a license on an unsupported platform, the operation succeeds, but the **show license** output indicates that the license is not supported.

In the Network OS v3.0.0 release, this command is supported only on the local RBridge.

## Examples

To add a license on the local switch:

```
switch# license add licstr "*B r84pNRtHKdRZujmwAUT63GORXIpBhBZK0ckRq6Bvv13Strvw1:fUjANF
av5W:gWx3hH2:9RsMv3BHfeCRFM2gj9N1krdIiBPBoa4xfSD2jf,Xx1RwksliX8fH6gpx7,73t#"
```

```
Adding license [*B r84pNRtHKdRZujmwAUT63GORXIpBhBZK0ckRq6Bvv13Strvw1:fUjANF
av5W:gWx3hH2:9RsMv3BHfeCRFM2gSLj9N1krdIiBPBoa4xfSD2jf,Xx1RwksliX8fH6gpx7,73t#]
```

To add a Dynamic Ports on Demand (DPOD) license on a switch that does not support the feature:

```
switch# license add licstr "*B
a6q3zwcUaNkWHPOfVf8afFZqHYype6sQxaEr5HIeFD3nba74i43BnRt6T8b2sDPtVMKuMfUPwV8NvHDXxFgbB3f2w3pJNlujxLVdIVkX
doNHf6i4SzwuvimIj0ORN:JOojLU#"

License Added [*B
a6q3zwcUaNkWHPOfVf8afFZqHYype6sQxaEr5HIeFD3nba74i43BnRt6T8b2sDPtVMKuMfUPwV8NvHDXxFgbB3f2w3pJNlujxLVdIVkX
doNHf6i4SzwuvimIj0ORN:JOojLU#]
switch# show license

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Ports on Demand license - not applicable on this platform license
Feature name:PORTS_ON_DEMAND_1
License is valid
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

## Related Commands

[license remove](#), [show license](#), [show license id](#)

# license remove

Removes a license key from a switch or deactivates a temporary license that cannot be removed..

## Syntax

```
license remove licstr { licenseString | feature } [ rbridge-id ID ]
```

## Command Default

This command is executed on the local switch.

## Parameters

**licstr** *licenseString*

Removes the specified license string and associated feature. The license string must be enclosed in double quotation marks.

**licstr** *feature*

Removes the license string associated with the specified feature from the license database of the local switch. The feature name must be enclosed in double quotation marks. Supported licensed features include the following: FCOE\_BASE, ADVANCED\_SERVICES, LAYER\_3, PORT\_10G\_UPGRADE and PORT\_40G\_UPGRADE, and PORT\_100G\_UPGRADE.

**rbridge-id** *ID*

Executes the command on the remote switch specified by the RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

You cannot display the license string once you install it. If you do not remember the string, use the feature name displayed in the **show license** command output to remove the license.

Depending on the feature being removed you must first clear all license-related configurations, and possibly disable and re-enable selected ports for this command to take effect. Follow the instructions in the command output.

This command deactivates but does not permanently remove time-based trial licenses.

You must disable or remove all configurations related to a licensed feature before you can remove the license for that feature. To remove the 10G, 40G, and 100G Port Upgrade licenses, you must remove all non-Base-allowance port reservations for the respective license type.

In the Network OS v3.0.0 release this command is supported only on the local RBridge.



## Examples

To remove a license string from the local switch:

```
switch# license remove licstr "*B r84pNRtHKdRZujmAUT63GORXIpBhBZK0ckRq6Bvvl3Strvwl:fUjANF  
av5W:gWx3hH2:9RsMv3BHfeCRFM2gSLj9NlkrdIiBPBoa4xfSD2jf,Xx1RwksliX8fH6gpx7,73t#
```

```
Removing license for rbridge-id 2 [*B r84pNRtHKdRZujmAUT63GORXIpBhBZK0ckRq6Bvvl3Strvwl:fUjANF  
av5W:gWx3hH2:9RsMv3BHfeCRFM2gSLj9NlkrdIiBPBoa4xfSD2jf,Xx1RwksliX8fH6gpx7,73t#]
```

To remove a license based on the feature name from the local switch:

```
switch# license remove licstr "FCOE_BASE"
```

```
removing license feature name [FCOE_BASE]
```

## Related Commands

[license add](#), [show license](#), [show license id](#)

# listen-limit

Sets a global limit of BGP dynamic subnet range neighbors.

## Syntax

`listen-limit max-num`

`no listen-limit`

## Command Default

The default listen limit value is 100.

## Parameters

*max-num*

Specifies the listen limit value. Enter an integer from 1 through 255.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

This command supports only IPv4 BGP.

When the global or peer level limit is increased, if any new connection comes in from remote end and falls under the range, it is accepted.

If the limit has been reached and you reduce the global or peer-group limit, the previously ESTABLISHED dynamic neighbors are not be destroyed. You must use the **clear neighbor** command.

When the new sessions are being created, then device uses the updated limit. If limit has been reached and a request for new connection is received, the connections are not accepted and the information is logged.

## Examples

The following example limits the number of dynamic neighbors that can be created to 150 globally.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# listen-limit 150
```

The following example restores the number of dynamic neighbors that can be created to the default value of 100.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# no listen-limit
```

## History

Release version	Command history
7.1.0	This command was introduced.

# listen-range

Associates a subnet range with a BGP peer group and sets the maximum number of BGP dynamic neighbors that can be created for this range.

## Syntax

**listen-range** *ip address/mask* **peer-group** *peer-group-name* [ **limit** *num* ]

**no listen-range** *ip address/mask* **peer-group** *peer-group-name*

## Command Default

Disabled.

## Parameters

*ip address/mask*

Specifies an IP address and network mask.

**peer-group** *peer-group-name*

Specifies a peer group.

**limit** *num*

Listen limit value. Enter an interger from 1 through 255.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

## Usage Guidelines

This command supports only IPv4 BGP. The **no** form of the command deletes the listen range and tears down all dynamic neighbor formed for this range.

## Examples

This example associates a subnet range of 10.1.0.0/16 with a peer group called "mypeergroup" and sets the maximum number of BGP dynamic neighbors that can be created to 80.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# listen-range 10.1.0.0/16 peer-group mypeergroup limit 80
```

## History

Release version	Command history
7.1.0	This command was introduced.

# line vty exec-timeout

Sets the CLI session timeout.

## Syntax

```
line vty exec-timeout timeout  
no line vty exec-timeout
```

## Command Default

The default timeout value is 10 minutes.

## Parameters

*timeout*

Specifies the CLI session timeout period in minutes. The timeout value specifies the amount of time a CLI session can be idle before it logs you out. Valid values range from 0 through 136.

## Modes

Global configuration mode

## Usage Guidelines

The **line vty exec timeout** command is a configuration command and the timeout value set by this command holds for subsequent login sessions, unless it is overwritten for a single session with the **terminal timeout** command. The terminal timeout command is not a configuration command and the timeout value set by this command controls only the current session. After the current session times out, the **line vty exec timeout** value applies for subsequent sessions.

This command is supported only on the local switch.

This command is not available on the standby management module.

Enter **no line vty exec-timeout** to disable auto-logout and delete the timeout value.

## Examples

To set the terminal timeout to 60 minutes:

```
switch(config)# line vty exec-timeout 60  
switch(config-line-vty)# exit  
switch(config)# exit  
switch# show running-config line vty  
  
line vty  
exec-timeout 60  
!
```

## Related Commands

[terminal](#)

# linecard

Configures a line card (interface module).

## Syntax

**linecard** *slot\_number linecard\_type*

**no linecard** *slot\_number*

## Parameters

*slot\_number*

Specifies the slot number to be configured. Line card slots are slots 1 through 4 on an Extreme VDX 8770-4 and slots 1 through 8 on an Extreme VDX 8770-8.

*linecard\_type*

Specifies the type of line card. Enter **linecard** *slot\_number linecard\_type ?* to display currently supported types.

## Modes

Global configuration mode

RBridge ID configuration mode

## Usage Guidelines

Use this command to configure the specified slot for an line card of a given type.

The command is executed in the context of the given RBridge. You must first enter the rbridge-id context for the specific line card. Once you are in the rbridge-id context, enter **linecard***slot\_number linecard\_type* to configure the slot. If you replace a given line card with another one of a different type, you must remove the configuration and then reconfigure the slot.

The line card must be powered off before you can remove the slot configuration.

The LC72x1G type displayed under "possible completion" is not supported.



### CAUTION

Enter **no linecard** to remove the slot configuration. When hot-swapping line cards of different types, copy the running-config file to the startup-config file before rebooting. This ensures that the desired changes are persistent in case there are any hardware or software incompatibilities.



## Examples

To configure a slot for an line card on a switch and to verify the configuration:

```
device# configure terminal
Entering configuration mode terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# linecard 1 ?
Possible completions:
  LC12x40G   12X40G linecard
  LC48x1G    48X1G linecard
  LC48x10G   48X10G linecard
  LC72x1G    72X1G linecard
device(config-rbridge-id-1)# linecard 1 LC48x10G
Creating new linecard configuration was successful.
device(config-rbridge-id-1)# do show running-config rbridge-id 1 linecard

rbridge-id 1
  linecard 1 LC48x10G
  linecard 4 LC48x10G
```

## Related Commands

[show running-config rbridge-id linecard](#)

# lldp dcbx-version

Specifies which version of the Data Center Bridging Exchange (DCBX) protocol to use.

## Syntax

```
lldp dcbx-version { auto | cee }  
no lldp dcbx-version
```

## Command Default

The default setting is **auto**.

## Parameters

### auto

Specifies to auto-adjust the DCBX protocol version to accommodate the difference when a switch interacts with different vendors using a different version of the DCBX protocol.

### cee

Specifies to use the Converged Enhanced Ethernet (CEE) DCBX version. The pre-CEE version is not available if this option is selected.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Devices enabled for data center bridging can use the DCBX protocol to discover and exchange information about their administratively configured capabilities. DCBX eliminates the need to configure a large number of switches in the network.

Enter **no lldp dcbx-version** to return to the default setting of **auto**.

## Examples

To specify that the CEE version be used on a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9  
switch(conf-if-te-178/0/9)# lldp dcbx-version cee
```

# lldp disable

Disables the Link Layer Discovery Protocol (LLDP) on the interface.

## Syntax

**lldp disable**

**no lldp disable**

## Command Default

LLDP is enabled at both the global and interface levels.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no lldp disable** to enable LLDP on a specific interface.

## Examples

To disable LLDP on a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# lldp disable
```

To enable LLDP on a specific 40-gigabit Ethernet interface:

```
switch(config)# interface fortygigabitethernet 1/3/1
switch(conf-if-fo-1/3/1)# no lldp disable
```

# lldp iscsi-priority

Sets the priority that will be advertised in the DCBX iSCSI TLV for a specified interface.

## Syntax

`lldp iscsi-priority value`

`no lldp iscsi-priority`

## Command Default

Priority value is 4.

## Parameters

*value*

Specifies the priority value. Valid values range from 0 through 7.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter `no lldp iscsi-priority` to return to the default setting.

## Examples

To set the iSCSI priority value to 5 on a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# lldp iscsi-priority 5
```

# lldp profile

Applies a Link Layer Discovery Protocol (LLDP) profile to an interface.

## Syntax

**lldp profile** *name*

**no lldp profile**

## Command Default

LLDP profile name.

## Parameters

*name*

Specifies the profile name. Valid profile name length is between 1 and 32 characters.

## Modes

Interface subtype configuration mode

## Usage Guidelines

You must use the **lldp profile** command to create an LLDP profile before you can apply the profile to the interface. Only one LLDP profile can exist at any time for a particular interface. When this command is not present, the parameters defined in the global LLDP configuration are used.

Enter **no lldp profile** to delete the profile from the interface.

## Examples

To apply an LLDP profile called *test* on an specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# lldp profile test
```

## Related Commands

[interface](#), [lldp dcbx-version](#), [lldp disable](#), [lldp iscsi-priority](#)

# load-balance

Configures load balancing settings.

## Syntax

```
load-balance [ dst-mac-vid | src-dst-ip | src-dst-ip-mac-vid | src-dst-ip-mac-vid-port | src-dst-ip-port | src-dst-mac-vid |
src-mac-vid ]
```

```
no load-balance
```

## Command Default

The default setting is the operand *src-dst-ip-mac-vid-port*, which means that source and destination IP, MAC address, VID and TCP/UDP port-based load balancing are used.

## Parameters

### dst-mac-vid

Specifies that destination MAC address and VID-based load balancing will be used.

### src-dst-ip

Specifies that source and destination IP address-based load balancing will be used.

### src-dst-ip-mac-vid

Specifies that source and destination IP and MAC address and VID-based load balancing will be used.

### src-dst-ip-mac-vid-port

Specifies that source and destination IP, MAC address, VID and TCP/UDP port-based load balancing will be used.  
This is the default.

### src-dst-ip-port

Specifies that source and destination IP and TCP/UDP port-based load balancing will be used.

### src-dst-mac-vid

Specifies that source and destination MAC address and VID-based load balancing will be used.

### src-mac-vid

Specifies that source MAC address and VID-based load balancing will be used.

## Modes

Port-channel configuration mode

## Usage Guidelines

Use the **no** form of this command to return to the default setting.

When configuring load balancing on an Extreme VDX 6740, it should be configured consistently for all port-channels on the switch. These switches support one load-balancing scheme at a time, and apply the last loaded load-balancing scheme to all port-channels on the switch. This is not required for the Extreme VDX 8770 platform, as it supports multiple port-channel load-balancing schemes.

## Examples

To set load balancing to use the destination MAC address and VID-based load balancing:

```
device# configure
device(config)# interface port-channel 10
device(config-Port-channel-10)# load balance dst-mac-vid
```

# load-balancing

Configures load balancing on an RBridge.

## Syntax

```
load-balancing threshold-priority threshold-priority-value
no load-balancing
```

## Command Default

None

## Parameters

**threshold-priority** *threshold-priority-value*  
The load balancing threshold priority. The range is from 1 through 254.

## Modes

Fabric-Virtual-Gateway on an RBridge VE interface IPv4 or IPv6 configuration mode

## Usage Guidelines

Enter the **no** form of the command to remove the threshold priority value on an RBridge.

## Examples

The following example shows how to configure load balancing.

```
switch(config)# rbridge-id 1
switch(config-rbridge-id-1)# interface ve 2000
switch(config-Ve-2000)# ip fabric-virtual-gateway 23
switch(config-ip-fabric-virtual-gw)# load-balancing threshold-priority 100
```

## History

Release version	Command history
5.0.1	This command was introduced.



# load-balancing-disable

Disables load balancing globally.

## Syntax

`load-balancing-disable`

`no load-balancing-disable`

## Command Default

Load balancing is enabled globally.

## Modes

Fabric-Virtual-Gateway in VE interface IPv4 or IPv6 configuration mode

## Usage Guidelines

Enter the **no** form of the command to re-enable load balancing globally.

## Examples

The following example shows how to disable load balancing.

```
switch(config)# interface ve 2000
switch(config-ve-2000)# ip fabric-virtual-gateway
switch(config-ip-fabric-virtual-gw)# load-balancing-disable
```

## History

Release version	Command history
5.0.1	This command was introduced.

# local-as

Specifies the BGP autonomous system number (ASN) where the device resides.

## Syntax

**local-as** *num*

**no local-as** *num*

## Parameters

*num*

The local ASN. The range is from 1 through 4294967295.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the ASN from the device.

ASNs in the range from 64512 through 65535 are private numbers that are not advertised to the external community.

BGP address-family IPv6 unicast VRF configuration mode is not supported. The IPv6 configuration is inherited from an IPv4 configuration.

## Examples

This example assigns a separate local AS number.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# local-as 777
```

This example assigns a separate local AS number for BGP VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# router bgp
device(config-bgp-router)# local-as 200
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf) local-as 500
```

## History

Release version	Command history
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# log (OSPFv2)

Controls the generation of OSPFv2 logs.

## Syntax

```
log { adjacency | all | bad-packet [ checksum ] | database | retransmit }
```

```
no log { adjacency | all | bad-packet [ checksum ] | database | retransmit }
```

## Command Default

Disabled. Only OSPFv2 messages indicating possible system errors are logged.

## Parameters

### adjacency

Specifies the logging of essential OSPFv2 neighbor state changes.

### all

Specifies the logging of all syslog messages.

### bad-packet

Specifies the logging of bad OSPFv2 packets.

### checksum

Specifies all OSPFv2 packets that have checksum errors.

### database

Specifies the logging of OSPFv2 LSA-related information.

### retransmit

Specifies the logging of OSPFv2 retransmission activities.

## Modes

OSPF router configuration mode

OSPF VRF router configuration mode

## Usage Guidelines

Use this command to disable or re-enable the logging of specific events related to OSPFv2. If this command is not enabled only OSPFv2 messages indicating possible system errors are logged.

Use the **no** form of this command to restore the default.

## Examples

This example enables the logging of all OSPFv2-related syslog events.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router ospf
device(config-router-ospf-vrf-default-vrf)# log all
```

This example enables the logging of OSPFv2 retransmission activities.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router ospf
device(config-router-ospf-vrf-default-vrf)# log retransmit
```

## History

Release version	Command history
6.0.0	This command was introduced.

# log (OSPFv3)

Controls the generation of OSPFv3 logs.

## Syntax

```
log { adjacency [ dr-only ] | all | bad-packet [ checksum ] | database | retransmit }
```

```
no log { adjacency | all | bad-packet [ checksum ] | database | retransmit }
```

## Command Default

Only OSPFv3 messages indicating possible system errors are logged.

## Parameters

### adjacency

Specifies the logging of essential OSPFv3 neighbor state changes.

### dr-only

Specifies the logging only of designated router (DR) interface adjacency changes.

### all

Specifies the logging of all syslog messages.

### bad-packet

Specifies the logging of bad OSPFv3 packets.

### checksum

Specifies all OSPFv3 packets that have checksum errors.

### database

Specifies the logging of OSPFv3 LSA-related information.

### retransmit

Specifies the logging of OSPFv3 retransmission activities.

## Modes

OSPFv3 router configuration mode

OSPFv3 VRF router configuration mode

## Usage Guidelines

Use this command to disable or re-enable the logging of specific events related to OSPFv3. If this command is not enabled, only OSPFv3 messages indicating possible system errors are logged.

The **no** form of the command restores the default.

## Examples

The following example enables the logging of all OSPFv3-related syslog events.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# log all
```

The following example enables the logging of OSPFv3 retransmission activities.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# log retransmit
```

## History

Release version	Command history
6.0.1	This command was introduced.

# log-dampening-debug

Logs dampening debug messages.

## Syntax

`log-dampening-debug`

`no log-dampening-debug`

## Command Default

This option is disabled.

## Modes

BGP configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

## Examples

The following example logs dampening debug messages.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# log-dampening-debug
```

# logging auditlog class

Sets the severity levels (class) for the audit log class.

## Syntax

**logging auditlog class** *class*

**no logging auditlog class** *class*

## Command Default

CONFIGURATION, FIRMWARE, and SECURITY audit log classes are enabled.

## Parameters

*class*

Specifies the class name of the audit log. Valid classes are CONFIGURATION, FIRMWARE, and SECURITY.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on the local switch.

This command is not supported on the standby management module.

The total message storage available is 2048 messages.

Enter **no logging auditlog class** *class* to remove the audit logging for the specified class.

## Related Commands

[clear logging auditlog](#), [clear logging raslog](#)



# logging raslog console

Sets the severity levels for the RASLog console and allows users to temporarily stop showing RASLog messages on the console.

## Syntax

`logging raslog console severity`

`no logging raslog console severity`

`logging raslog console stop [ minutes ]`

## Command Default

Severity level is INFO.

## Parameters

*severity*

Specifies the minimum severity level of the message to pass through the filter. Valid values consist of one of the following: INFO, WARNING, ERROR, or CRITICAL. Input values are case-sensitive.

**start**

Initiates RASLog messages.

**stop**

Stops RASLog messages.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on the local switch.

This command is not supported on the standby management module.

The total message storage available is 2048 messages.

When stopping or starting RASLog messages, the commands are not persistent and therefore are not configuration commands.

If the command `logging raslog console stop minutes` is invoked before the previous time value expires, the latest CLI duration applies.

## Examples

To reset the RASLog severity levels to the default value.

```
switch(config)# no logging raslog console
```

To stop RASLog messages for 1 minute:

```
switch# logging raslog console stop 1  
Logging message have been blocked on console for 1 minutes
```

To start RASLog messages:

```
switch# logging raslog console start
```

```
2013/11/14-08:42:57, [RAS-3008], 5348, M2 | Active, INFO, VDX8770-4, Logging messages to console has  
been reset by user.
```

## Related Commands

[clear logging raslog](#), [show running-config logging](#)

# logging syslog-client

Configures a switch to forward system messages to specified client IP addresses.

## Syntax

```
logging syslog-client localip {MM_IP | CHASSIS_IP }
```

## Parameters

### MM\_IP

Directs the system messages to the Management Module IP address.

### CHASSIS\_IP

Directs the system messages to the IP address assigned to the chassis.

## Modes

Global configuration mode

## Usage Guidelines

This command is not supported on the standby management module.

## Examples

Example command for directing the logging data to the management module IP address.

```
device(config)# logging syslog-client localip MM_IP
```

# logging syslog-facility local

Configures the syslog facility.

## Syntax

```
logging syslog-facility local log_level
```

## Command Default

Syslog level is LOG\_LOCAL7.

## Parameters

*log\_level*

Specifies the syslog facility level. Valid log levels include the following: LOG\_LOCAL0, LOG\_LOCAL1, LOG\_LOCAL2, LOG\_LOCAL3, LOG\_LOCAL4, LOG\_LOCAL5, LOG\_LOCAL6, LOG\_LOCAL7

## Modes

Global configuration mode

## Usage Guidelines

Use this command to configure the log level for all error log entries to forward to one or more specified servers. You can configure up to four servers.

When used without a log level parameter, use this command to display the current value.

This command is supported only on the local switch.

This command is not supported on the standby management module.

## Examples

To configure the syslog facility level:

```
switch(config)# logging syslog-facility local LOG_LOCAL5
```

## Related Commands

[logging syslog-server](#), [show running-config logging syslog-server](#)

# logging syslog-server

Configures a device to forward system messages to specified servers.

## Syntax

```
logging syslog-server ip_address [ secure ] [ port port-num ] [ use-vrf vrf-name ]
no logging syslog-server ip_address [ secure ] [ port port-num ] [ use-vrf vrf-name ]
```

## Parameters

*ip\_address*

Specifies the IP address of the syslog server in IPv4 or IPv6 format.

**secure**

Configures a secure default (port 514) or specified nondefault syslog server port. A secure port number with default values is not shown in the Extreme Network OS database.

**port** *port-num*

Specifies a nondefault port. The port range is from 1 through 65535.

**use-vrf** *vrf-name*

Specifies a VRF through which to communicate with the server. See the Usage Guidelines.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to configure a device to forward all error log entries to the one or more specified servers. You can configure up to four servers.

A secure port number with default values is not shown in the Extreme Network OS database.

The **crypto import syslogca** command is required for secure syslog to be fully functional.

You can configure up to four syslog servers; this includes all VRFs. You must execute the command for each server.

This command is not supported on the standby management module.

In an Extreme VCS Fabric, the syslog configuration is distributed to all devices in the fabric.

Use the **no logging syslog-server** command with the optional **use-vrf** keyword to remove the specified IP address VRF.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

To configure a server IPv4 address to which system messages are sent on a user-specified VRF:

```
device(config)# logging syslog-server 192.168.163.233 use-vrf myvrf
device(config-syslog-server-192.168.163.233/myvrf)#
```

To configure a server IPv4 address and specify a VRF with a secure nondefault port, and confirm the configuration:

```
device(config)# logging syslog-server 192.168.163.233 use-vrf myvrf secure port 1999
device(config-syslog-server-192.168.163.233/myvrf)# do show running-config logging syslog-server
logging syslog-server 192.168.163.233 use-vrf myvrf
secure port 1999
```

To remove a configured syslog server:

```
device(config)# no logging syslog-server 192.168.163.233
```

To remove a syslog nondefault server port and confirm the configuration:

```
device(config)# no logging syslog-server 10.17.17.203 secure port 1999
device(config)# do show running-config logging syslog-server
logging syslog-server 10.17.17.203
secure
```

## History

Release version	Command history
6.0.1	This command was modified to support the <b>use-vrf</b> keyword.
7.0.0	This command was modified to support the <i>vrf-name</i> variable.

## Related Commands

[certutil import syslogca](#), [logging syslog-facility local](#), [show running-config logging syslog-facility](#), [show running-config logging syslog-server](#)

# logical-chassis principal-priority

Sets the priority of a switch to assign a specific RBridge ID the role of principal node in a logical chassis cluster.

## Syntax

```
logical-chassis principal-priority priority-value
```

```
no logical-chassis principal-priority
```

## Parameters

*priority-value*

Sets the priority for the switch. A lower number means a higher priority. Values range from 1 through 128.

## Modes

RBridge ID configuration mode

## Usage Guidelines

If all switches boot up at the same time, the default priority is the same and all switches will compare their mutual intents. The switch with the lowest switch WWN becomes the principal switch. However, you can use this command to select the principal switch in a logical chassis cluster. For this command to take effect, you need to issue the **logical-chassis principal-switchover** command.

This command can be used only on nodes that are part of a logical chassis cluster. The node, however, can be disconnected from the cluster when you issue the command.

Use the **no** form of this command to remove a priority value from this node.

You can view the principal priority in both the **show running config** (using the **rbridge-id** operand) and **show vcs detail** command outputs (both are run in Privileged EXEC mode).

## Examples

To set the principal priority to 5 for switch that is in logical chassis cluster:

```
switch# configure
switch(config)# rbridge-id 5
switch(config-rbridge-id-5)# logical-chassis principal-priority 5
```

## Related Commands

[logical-chassis principal-switchover](#)

# logical-chassis principal-switchover

Triggers a fabric reformation and elects a principal node based on the principal priority value.

## Syntax

```
logical-chassis principal-switchover
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Issue this command after you have used the **logical-chassis principal-priority** *priority-value* command so that the priority you set takes effect and a new principal node is selected on the cluster.

## Examples

```
switch# configure
switch(config)# rbridge-id 5
switch(config-rbridge-id-5)# logical-chassis principal-priority 1
switch(config-rbridge-id-5)# end
switch# logical-chassis principal-switchover
```

## Related Commands

[logical-chassis principal-priority](#)



# long-distance-isl

Extends an ISL link up to 30 km.

## Syntax

```
long-distance-isl { 2000 | 5000 | 10000 | 30000 }
no long-distance-isl
```

## Command Default

The default is 2 km.

## Parameters

**2000**

Specifies a 2 km distant link.

**5000**

Specifies a 5 km distant link.

**10000**

Specifies a 10 km distant link.

**30000**

Specifies a 30 km distant link. DCB/FCoE capabilities are not supported with this setting.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Metro VCS supports long-distance ISL ports up to 30 km on the Extreme VDX platforms listed below. Links up to 10 km are lossless. You can have eight 1-km links forming an Extreme trunk. You can also have mixed length cables forming the ISL. For ECMP purposes, you can have eight 8-link ECMP trunks.

**TABLE 7** Limitations for long-distance Metro VCS

Supported hardware	Extended ISL up to 2 km	Extended ISL up to 5 km	Extended ISL up to 10 km	Extended ISL up to 30 km
Extreme VDX 6740	yes	yes	yes	yes
Extreme VDX 8770 - VDX LC48x10G line card	yes	yes	yes	yes

The following displays the limitations on extended ISL for Network OS hardware.

**TABLE 8** Conditions for long distance Metro VCS

Condition	Extended ISL up to 2 km	Extended ISL up to 5 km	Extended ISL up to 10 km	Extended ISL up to 30 km
Support for lossless FCoE/iSCSI traffic on the Metro VCS port-group	yes	yes	yes	no
Layer 2/IP Lossy Traffic support	yes	yes	yes	yes
Number of Metro VCS long distance ports supported per port group	1	1	1	1
Number of regular ISLs supported on a port group configured for long distance	1	1	0	0
Trunking support between multiple LD ISLs	no	no	no	no
CEE map or FCoE port allowed in same port-group	no	no	no	no
eNS Sync (MAC address table sync)	yes	yes	yes	yes
Zoning	yes	yes	yes	yes
HA failover	yes	yes	yes	yes
Node redundancy check	yes	yes	yes	yes
vMotion	yes	yes	yes	yes
Maximum PFCs Supported	3 (2 on the Extreme VDX 6740)	3 (2 on the Extreme VDX 6740)	3 (2 on the Extreme VDX 6740)	3 (2 on the Extreme VDX 6740)
Long-distance ISL on 40G to 4x10G breakout interfaces	no	no	no	no
Long-distance ISL on 1G and 10G copper interfaces	no	no	no	no

The following displays the port groups and number of port groups available on each platform for long distance Metro VCS.

**TABLE 9** Long distance Metro VCS port-group schema

Platform	Port groups	Number of port groups on platform
Extreme VDX 6740	1-32, 33-48 (49-50 do not support long distance)	2*
Extreme VDX 8770 (VDX LC48x10G linecard)	1-8, 9-16, 17-24, 25-32, 33-40, 41-48	6 per LCX10G blade

\*Not a valid deployment scenario at distances longer than 5 km, as no normal ISLs are allowed if both port-groups are configured with long-distance ISLs for 10 km and 30 km. For a 10 km ISL link, no other ISL links are allowed on the same ASIC.

For 2 km and 5 km ISL links, another short distance ISL link can be configured.

A maximum of three PFCs can be supported on a long distance ISL link.

Enter **no long-distance-isl** to revert to the default value.

## Examples

To extend the support of an ISL port with PFC by a distance of 5 km on a specific 10-gigabit Ethernet interface:

```
device(config)# interface tengigabitethernet 178/0/9
```

```
device(conf-if-te-178/0/9)# long-distance-isl 5000
```

## Related Commands

[interface](#), [isl-r\\_rdy](#)

## mac

Allows the user to add a MAC address to a MAC address group in a service or transport VF configuration supporting multitenancy in a Virtual Fabrics context.

## Syntax

```
mac mac_address
```

```
no mac mac_address
```

## Parameters

*mac\_address*

Specifies a MAC address in dot-separated hexadecimal notation.

## Modes

MAC group configuration mode

## Usage Guidelines

Use this command in MAC group configuration mode to add a MAC address to a MAC address group in a service or transport VF configuration supporting multitenancy in a Virtual Fabrics context.

Enter the MAC group configuration mode by using the **mac group** *mac-group-id* global configuration command.

Enter **no mac** *mac\_address* to remove a MAC addresses from the group.

### NOTE

You can add or remove only one MAC address per line.

## Examples

To enter MAC group configuration mode and add a MAC address to the group:

```
switch(config)# mac-group 1  
switch(config-mac-group 1)# mac abc1.abc2.abc3
```

To remove a MAC address from the group:

```
switch(config-mac-group 1)# no mac abc1.abc2.abc3
```

## Related Commands

[mac-group](#), [vcs virtual-fabric enable](#)

## mac access-group

Applies rules specified in a MAC access control list (ACL) to traffic entering or exiting an interface.

### Syntax

```
mac access-group ACLname { in | out } [ switched | routed ]
```

```
no mac access-group ACLname { in | out } [ switched | routed ]
```

### Parameters

*ACLname*

Specifies the name of the standard or extended MAC access list.

**in**

Specifies to filter inbound packets only.

**out**

Specifies to filter outbound packets only.

**switched**

Filter only switched traffic. This parameter is not valid for management or overlay-gateway interfaces.

**routed**

Filter only routed traffic. This parameter is not valid for management or overlay-gateway interfaces.

### Modes

Interface-subtype configuration mode

### Usage Guidelines

You can apply a maximum of six ACLs to a user interface, as follows:

- One ingress MAC ACL—if the interface is in switchport or overlay-gateway mode
- One egress MAC ACL—if the interface is in switchport mode
- One ingress IPv4 ACL
- One egress IPv4 ACL
- One ingress IPv6 ACL
- One egress IPv6 ACL

You can apply an ACL to multiple interfaces. And you can apply an ACL twice—ingress and egress—to a given user interface.

If you do not specify **switched** or **routed**, the ACL applies both to switched and routed traffic.

To remove an ACL from an interface, enter the **no** form of this command.

## Examples

The following example applies a MAC ACL named macacl2 to filter inbound packets only, on a specific ten-gigabit Ethernet interface.

```
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# mac access-group macacl2 in
```

The following example removes a MAC ACL from a specified port-channel interface.

```
device(config)# interface port-channel 62
device(config-Port-channel-62)# no mac access-group macacl2 in
```

# mac access-list standard

Creates a standard MAC access control list (ACL). Standard ACLs contain rules that permit or deny traffic based on source addresses that you specify.

## Syntax

```
mac access-list standard ACLname
```

```
no mac access-list standard ACLname
```

## Parameters

*ACLname*

Specifies an ACL name unique among all ACLs (Layer 2 and Layer 3). The name can be up to 63 characters in length, and must begin with an alphanumeric character. No special characters are allowed, except for the underscore and hyphen.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to create a standard MAC access list. If ACL is already created, this command puts the device in the standard MAC access-list configuration mode.

To remove a MAC ACL from an interface, enter the **no** form of this command.

## Examples

The following command creates a MAC standard ACL named mac1.

```
device(config)# mac access-list standard mac1  
device(conf-macl-std)#
```

The following command deletes a MAC standard ACL named mac1.

```
device(conf-macl-std)# no mac access-list standard mac1
```

# mac-address-reduction

Enables or disables the MAC address reduction feature.

## Syntax

```
mac-address-reduction [ enable | disable ]
```

## Parameters

**enable**

Enables the MAC address reduction feature.

**disable**

Disables the MAC address reduction feature.

## Modes

Protocol Spanning Tree configuration mode



# mac-address-table

Sets the aging time or adds static addresses to the MAC address table, and enables conversational MAC (address) learning.

## Syntax

```
mac-address-table { aging-time seconds | conversational aging_time | learning-mode conversational }
mac-address-table static mac-addr forward { <N>gigabitethernet rbridge-id/slot/port | port-channel number | vlan vlan_id }
no mac-address-table
no mac-address-table learning-mode
no mac-address-table static
```

## Command Default

Default aging time is 300 seconds.

Conversational MAC learning is disabled.

## Parameters

### aging-time *seconds*

Specifies the time in seconds that a learned MAC address will persist after the last update. If the aging time is set to zero (0), it means that aging is disabled. For Extreme VCS Fabric mode, values range from 60 through 100000.

### conversational *aging\_time*

Configures an aging time for conversational MAC addresses learned by destination address (DA) on an RBridge. If the aging time is set to zero (0), it means that aging is disabled. For Extreme VCS Fabric mode, values range from 60 through 100000.

### learning-mode conversational

Enables conversational MAC learning on an RBridge.

### static *mac-addr* forward

Specifies the Media Access Control (MAC) address (unicast or multicast) to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface.

### forward

Forwards the MAC address to the interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies the port-channel number. Valid values range from 1 through 63.

**vlan** *vlan\_id*

Specifies an active VLAN. Range is from 1 through 4090 if Virtual Fabrics is disabled, and 1 through 8191 if Virtual Fabrics is enabled.

## Modes

Global configuration mode

## Usage Guidelines

The **vlan** keyword is mandatory because the switch only supports independent VLAN learning (IVL).Enter **no mac-address-table** to reset the values to their defaults.Enter **no mac-address-table learning-mode** to disable conversational MAC learning on an RBridge.

## Examples

To add the static address 0011.2222.3333 to the MAC address table with a packet received on VLAN 100:

```
switch(config)# mac-address-table static 0011.2222.3333 forward tengigabitethernet 0/1 vlan 100
```

To set the aging time to 10 minutes:

```
switch(config)# mac-address-table aging-time 600
```

To set the aging time to 10 minutes for conversational MAC addresses:

```
switch(config)# mac-address-table aging-time conversational 600
```

To enable conversational MAC learning:

```
switch(config)# mac-address-table learning-mode conversational
```

To disable the static aging time:

```
switch(config)# no mac-address-table aging-time static
```

To disable the conversational aging time:

```
switch(config)# no mac-address-table aging-time conversational
```

To disable static MAC address forwarding on an Ethernet interface:

```
switch(config)# no mac-address-table static aaaa.bbbb.cccc forward tengigabitethernet 1/0/1
```

To disable the aging time by setting its value to 0:

```
switch(config)# mac-address-table aging-time 0
```

## Related Commands

[show mac-address-table](#)

# mac-address-table consistency-check interval

Specifies the time interval for the MAC consistency-check trigger, across multiple nodes. After an interval specified by *seconds*, consistency check detects any MACs that are not synchronized and resynchronizes them.

## Syntax

`mac-address-table consistency-check interval seconds`

`no mac-address-table consistency-check interval`

## Command Default

If consistency-check is not suppressed—but the consistency-check interval is not specified—consistency check uses the default value of 300 seconds.

## Parameters

*seconds*

Specifies the MAC consistency-check interval, in seconds. Valid values are 120 through 3600. The default value is 300.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of this command restores the consistency-check interval to 300 seconds.

## Examples

The following example sets 500 seconds as the consistency-check interval.

```
device# configure terminal
device(config)# mac-address-table consistency-check interval 500
```

The following example resets the consistency-check interval to the default 300 seconds.

```
device# configure terminal
device(config)# no mac-address-table consistency-check interval
```

## History

Release version	Command history
6.0.0	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[mac-address-table consistency-check suppress](#), [show mac-address-table consistency-check](#)

# mac-address-table consistency-check suppress

Suppresses MAC-address consistency check and resynchronization on the switch or VCS.

## Syntax

```
mac-address-table consistency-check suppress
no mac-address-table consistency-check suppress
```

## Command Default

If consistency-check/resynchronization is not suppressed by this command, it is enabled by default.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command restores enablement of MAC-address consistency-check and resynchronization.

## Examples

The following example disables consistency-check/resynchronization on the device.

```
device# configure terminal
device(config)# mac-address-table consistency-check suppress
```

## History

Release version	Command history
6.0.0	This command was introduced.

## Related Commands

[mac-address-table consistency-check interval](#), [show mac-address-table consistency-check](#)

# mac-address-table mac-move action

Specifies an action to be taken when the MAC-move detection algorithm identifies a port of interest.

## Syntax

```
mac-address-table mac-move action { shutdown | raslog }
```

```
no mac-address-table mac-move action [ shutdown | raslog ]
```

## Command Default

This feature is not enabled. See also the Usage Guidelines.

## Parameters

### shutdown

Causes a selected port to be shut down and a RASLog to be generated regarding port, MAC address, and VLAN status.

### raslog

Causes a RASLog to be generated regarding port, MAC address, and VLAN status, without shutting down the port.

## Modes

Global configuration mode

## Usage Guidelines

The default action taken when a port of interest is identified is **shutdown**.

Use the **no** form of this command to disable either or both options.

Use the **show mac-address-table mac-move** command to display the results of this command.

## Examples

To enable a port of interest to be shut down and a RASLog to be generated:

```
device(config)# mac-address-table mac-move action shutdown
```

To disable the generation of a RASLog even though a port of interest is not shut down:

```
device(config)# no mac-address-table mac-move action raslog
```

## History

Release version	Command history
7.1.0	This command was introduced.

## Related Commands

[show mac-address-table mac-move](#)



# mac-address-table mac-move auto-recovery

Enables auto-recovery and specifies an interval for auto-recovery following a MAC-move detection and response.

## Syntax

```
mac-address-table mac-move auto-recovery { enable | time time }  
no mac-address-table mac-move auto-recovery enable  
no mac-address-table mac-move auto-recovery time
```

## Command Default

This feature is disabled. See also the Usage Guidelines.

## Parameters

### enable

Enables auto-recovery.

### time *time*

Specifies the interval, in minutes, between MAC-move response and recovery. Range is 3 through 30. The default is 5.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no mac-address-table mac-move auto-recovery enable** command to disable auto-recovery. (The **enable** keyword appears only if this feature has been enabled.)

Use the **no mac-address-table mac-move auto-recovery time** command to restore the default time.

Use the **show mac-address-table mac-move** command to display the results of this command.

## Examples

To enable auto-recovery:

```
device(config)# mac-address-table mac-move auto-recovery enable
```

To specify a time interval of 10 minutes:

```
device(config)# mac-address-table mac-move auto-recovery time 10
```

To disable auto-recovery:

```
device(config)# no mac-address-table mac-move auto-recovery enable
```

To restore the default time interval:

```
device(config)# no mac-address-table mac-move auto-recovery time
```

## History

Release version	Command history
7.1.0	This command was introduced.

## Related Commands

[show mac-address-table mac-move](#)

# mac-address-table mac-move detect

Repeated MAC-address moves, often caused by loops, overload control-plane resources. Enabling MAC-move detection by means of this command can resolve this problem in a VCS overlay topology.

## Syntax

```
mac-address-table mac-move detect
```

```
no mac-address-table mac-move detect
```

## Command Default

This feature is disabled.

When this feature is enabled, the default number of MAC-moves that are detected is 20. This limit can be changed by means of the `mac-address-table mac-move limit` command.

## Modes

Global configuration mode

## Usage Guidelines

The `no` form of this command disables MAC-move detection.

## Examples

The following example enables MAC-move detection on the switch.

```
device# configure terminal
device(config)# mac-address-table mac-move detect
```

## History

Release version	Command history
6.0.0	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[mac-address-table mac-move limit](#), [show ip interface](#), [show ipv6 interface](#), [show mac-address-table mac-move](#)

# mac-address-table mac-move limit

Specifies the upper limit for MAC-address-moves detected—in any 10-second window—without triggering MAC-address-move resolution.

## Syntax

```
mac-address-table mac-move limit move_threshold
```

```
no mac-address-table mac-move limit
```

## Command Default

When MAC-address-move detection is enabled, by means of the **mac-address-table mac-move detect** command, and *move\_threshold* is not specified, the default for *move\_threshold* is 20.

## Parameters

*move\_threshold*

Specifies the number of MAC-address moves (in any 10-second window) above which the repeated-MAC-moves feature is triggered. Range is from 5 through 500. The default is 20.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command sets *move\_threshold* to 20, which triggers the feature from the first MAC-address move.

## Examples

The following example sets the number of MAC-moves detected without triggering MAC-address-move resolution to 10.

```
device# configure terminal
device(config)# mac-address-table mac-move limit 10
```

The following example resets the *move\_threshold* to zero.

```
device# configure terminal
device(config)# no mac-address-table mac-move limit
```

## History

Release version	Command history
6.0.0	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[mac-address-table mac-move detect](#), [show ip interface](#), [show ipv6 interface](#), [show mac-address-table mac-move](#)

## mac-group

Creates a MAC address group into which one or more end-station MAC addresses are defined, supporting service or transport VFs in a Virtual Fabrics context. The group is used in MAC-based VLAN classification at the access port.

### Syntax

```
mac-group mac-group-id
no mac-group mac-group-id
```

### Parameters

*mac-group-id*  
A fabric-wide ID. Values range from 1 through 500.

### Modes

Global configuration mode

### Usage Guidelines

Use this command to enter MAC group configuration mode. In that mode, use the **mac** command to enter one or more MAC addresses that become members of the group.

Enter **no mac-group***mac-group-id* to delete the group and all MAC addresses associated with it.

#### NOTE

You can add or remove only one MAC address per line.

### Examples

To enter MAC group configuration mode and add a MAC address to the group:

```
switch(config)# mac-group 1
switch(config-mac-group 1)# mac abc2.abc2.abc2
```

To remove a MAC address from the MAC group:

```
switch(config-mac-group 1)# no mac abc1.abc2.abc3
```

To remove a MAC group and its associated MAC addresses:

```
switch(config)# no mac-group 1
```

### Related Commands

[mac](#), [vcs virtual-fabric enable](#)

# mac-learning disable vlan

Disables MAC address learning on an interface for specified VLANs.

## Syntax

```
mac-learning disable vlan { add | remove } { vlan vlan_id }
no mac-learning disable vlan
```

## Command Default

Dynamic MAC address learning is enabled.

## Parameters

### add

Adds a VLAN or range of VLANs to the list of VLANs for which dynamic MAC address learning is disabled.

### remove

Adds a VLAN or range of VLANs to the list of VLANs for which dynamic MAC address learning is disabled.

### vlan *vlan\_id*

Specifies a VLAN or range of VLANs. 802.1Q VLANs range from 1 through 4090. Extended VLANs in a Virtual Fabrics context range from 4096 through 8191.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no mac-learning disable vlan** command to enable dynamic MAC address learning for all VLANs on an interface .

Note the following supported configurations and limitations:

- This command is available on all switch ports.
- This command is not available on router ports or virtual routing interfaces. Appropriate error messages will be displayed.
- If this command is configured on a port channel (vLAG), dynamic MAC address learning is disabled on all the member ports. The configuration of this command on the members of the vLAG must be done individually, consistently, and uniformly.
- This command is not available on Inter-Switch Links (ISLs). MAC address learning is always disabled on ISLs. Appropriate error messages will be displayed.
- Source MAC address learning is not supported on VXLAN tunnel interfaces.
- This command is not allowed on a switch port that is attached to a switched virtual interface (SVI). The creation of the SVI will fail if MAC address learning is disabled on any interface that is part of the respective VLAN. Also, ARP resolution is affected if dynamic MAC learning is disabled on a switch port that is associated with a virtual routing interface (SVI).

- This command is disabled for the following VLANs, and appropriate error messages are displayed:
  - 1002 (FCoE VLAN)
  - 4093 (IP over TRILLVLAN)
  - 4095 (control VLAN)
- With CML support, destination MAC address learning is enabled on the switch. Disabling source MAC address learning does not have an effect on destination MAC address learning; however, the same MAC address appearing as a destination MAC address on other ports will trigger flooding.

## Examples

To disable dynamic MAC learning on VLAN 10:

```
switch(conf-if-te-4/0/5)# mac-learning disable vlan add 10
```

To disable dynamic MAC learning on VLANs 10 through 20:

```
switch(conf-if-te-4/0/5)# mac-learning disable vlan add 10-20
```

To enable dynamic MAC learning on VLAN 10:

```
switch(conf-if-te-4/0/5)# mac-learning disable vlan remove 10
```

To enable dynamic MAC learning on all VLANs:

```
switch(conf-if-te-4/0/5)# no mac-learning disable
```

## History

Release version	Command history
5.0.0	This command was introduced.



# mac-learning protocol bgp

By default, MAC address learning is enabled on VXLAN Layer 2 extension tunnels. This command delegates the responsibility for MAC learning on a tunnel to the Layer 3 control-plane protocol, such as BGP EVPN.

## Syntax

```
mac-learning protocol bgp
```

```
no mac-learning protocol bgp
```

## Command Default

Disabled.

## Modes

VXLAN overlay-gateway site configuration mode

## Usage Guidelines

BGP routing must be enabled. BGP EVPN must be configured.

The **no** form of the command disables BGP MAC learning and returns to the default of Layer 2 learning.

## Examples

This example changes the default MAC learning from Layer 2 to BGP MAC learning.

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# site mysite
device(config-overlay-gw-gateway1-site-mysite)# mac-learning protocol bgp
```

This example restores the default of Layer 2 learning.

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# site mysite
device(config-overlay-gw-gateway1-site-mysite)# no mac-learning protocol bgp
```

## History

Release version	Command history
7.0.0	This command was introduced.

# mac-rebalance

Forces the rebalancing of EXM entries for the MAC tables.

## Syntax

```
mac-rebalance port-channel number { rbridge-id rbridge-id }
```

## Parameters

**port-channel** *number*

Specifies the port-channel interface number. Valid values range from 1 through 6144.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Run this command on all remote (non-vLAG) nodes.

To achieve complete utilization of the entire vLAG member links, MAC entries learnt on vLAG need to be equally distributed among the vLAG member nodes. There are some scenarios, in which the EXM entries may not be balanced equally among the vLAG member nodes.

This command is applicable to remote RBridge nodes, such as non-vLAG member nodes. However there are not any restrictions on the usage of this command in vLAG member nodes.

Currently, EXM entries are balanced among the member nodes during RBridge membership changes (add or delete). MACs learned on vLAG are not rebalanced when the link updates (such as during LAG member additions or deletions), to avoid traffic disruption. However, when there are many link updates, the EXM mapping can become unbalanced and eventually overload the link capacity leading to frame drops. The mac-rebalance command corrects this scenario.

## Examples

This example rebalances the EXM entries on RBridge 1 (for vLAG 10):

```
switch# mac-rebalance port-channel 10 rbridge-id 1
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

# mac-refresh

Flushes MAC addresses on either the entire cluster or the partner edge loop-detection port.

## Syntax

```
mac-refresh interval { all | port }  
no mac-refresh
```

## Command Default

MAC flushing is disabled by default.

## Parameters

*interval*  
Specifies the number of seconds between MAC-addresss flushing.

**all**  
Flushes MAC addresses from the entire cluster.

**port**  
Flushes MAC addresses from the partner edge-loop-detection port.

## Modes

Edge loop detection mode.

## Usage Guidelines

Use the no form of this command to disable MAC-address flushing.

Use this command to remove any MAC inconsistencies in your system. If two interfaces are present in a layer-2 loop, each interface learns the same set of MAC addresses. When ELD detects the layer-2 loop, it puts the participating interface into an operationally down state. Consequently, MAC addresses learned on that interface get flushed. However, the same MAC addresses are present at the interface at the other end of the already detected loop, thereby creating this MAC inconsistency.

To remove this inconsistency, you can run the mac-refresh command to perform a MAC-flush on either the entire cluster or on the partner port at the other end of the loop.

## Examples

To flush all MAC addresses in the cluster every 150 seconds:

```
switch# configure  
switch (config)# protocol edge-loop-detection  
switch (conf-eld)# mac-refresh 150 all
```

## History

Release version	Command history
5.0.0	This command was introduced.

# management

Enables a variety of Dynamic Host Configuration Protocol (DHCP) management options.

## Syntax

```
management [ interface { autoconfig { dhcp | dhcpv6 } } ]  
no management
```

## Parameters

### interface

Enables management options.

### autoconfig

Enables automatic configuration of DHCP.

### dhcp

Enables DHCP for IPv4.

### dhcpv6

Enables DHCP for IPv6.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no** version of this command to disable this feature.

# map fport interface fcoe

Maps VF\_Ports to N\_Ports in Access Gateway (AG) mode and removes VF\_Port to N\_Port mapping.

## Syntax

```
map fport interface fcoe port
```

## Parameters

*port*

VF\_Port number

## Modes

N\_Port configuration mode

## Usage Guidelines

Use this command to specify a route that AG will use to direct traffic from a device (host or target) on a VF\_Port to a fabric switch port connected to an Access Gateway N\_Port. The process of specifying routes is called "mapping." Default mapping is enabled for the switch when enabling AG for the first time. You can use the **map** command to change the default mapping.

You must be in the configuration mode for the specific N\_Port where you want to map a VF\_Port. N\_Ports are designated by the format *rbridge-id/port group/N\_Port*, such as 3/0/4 for RBridge 3. Use this format to correctly identify the N\_Port in N\_Port configuration mode. VF\_Ports are identified by the format *domain/rbridge-id/VF\_Port*, such as 1/2/26.

## Examples

Map VF\_Port 1/2/26 to N\_Port 2/0/4.

```
switch(config-rbridge-id-2-ag-nport-if-fi-2/0/4)# map fport interface fcoe 1/2/26
```

Remove map from VF\_Port 1/2/26 to N\_Port 2/0/4.

```
switch(config-rbridge-id-2-ag-nport-if-fi-2/0/4)# no map fport interface fcoe 1/2/26
```

## Related Commands

[ag, nport interface fibrechannel \(port group\), pg, show ag map](#)

# map qos

Adds the QoS profile name as an action to the policy map.

## Syntax

```
map qos profile_name
```

## Parameters

*profile\_name*

Designates the name of the QoS profile to be added.

## Modes

Policy-map configuration mode

## Related Commands

[class](#), [policy-map](#)

# map sflow

Adds the sFlow profile name as an action to the policy map.

## Syntax

```
map sflow profile_name
```

## Parameters

*profile\_name*

Designates the name of the sFlow profile to be added.

## Modes

Policy-map configuration mode

## Examples

Typical command usage:

```
switch(config)# policy-map p1  
switch(config-policymap)# class c1  
switch(config-policyclass)# map sflow mysflowmap
```

## Related Commands

[class](#), [policy-map](#)



# map vlan

In a VXLAN overlay gateway configuration that uses Layer 2 extension, associates VLANs with VXLAN Network Identifiers (VNIs).

## Syntax

```
map vlan [ vlan_id ] {vni } [ vni ] [ auto ]
```

```
no map vlan vlan_id
```

```
no map vlan vni
```

## Parameters

*vlan\_id*

A single VLAN ID or range of VLAN IDs. The range is from 1 through 8191. See the Usage Guidelines.

**vni**

Specifies the VNI (VXLAN Network Identifier) token.

*vni*

A single VXLAN VNI or range of VXLAN VNIs. The range is from 1 through 16777215. See the Usage Guidelines.

**auto**

Enables automatic VLAN-to-VNI mapping for every VLAN associated with the tunnel.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

Note the following conditions:

- Before using this command, you must first set the VXLAN overlay gateway to layer2-extension, by means of the **type** command.
- Before using this command, you must first configure the appropriate VLANs to be used by the gateway.
- Before mapping VLANs to VNIs manually, you cannot have automatic mapping configured (by means of the **map vlan vni auto** command).
- You cannot map one VLAN to multiple VNIs. Similarly, you cannot map a single VNI to multiple VLANs. For example, vlan to vni mapping should be one to one.
- A single VLAN ID and a range of VLAN IDs can both be specified in a single command as follows: *x,y-z*. The same applies to VNIs.
- When using ranges, you must ensure that the number of values in a VLAN ID range corresponds to the number of values in a VNI range.
- The **no** forms of this command are allowed only if no VLANs are referenced by means of the **extend vlan** command (under a submode of the **site** command). For example, VLANs extended to a site should have a vni mapping.

- The **no map vlan vni auto** command disables the automatic assignment of VNIs. It is not allowed if manual VLAN-to-VNI mappings have been configured. For example, "auto" vlan to vni mapping and "explicit" vlan to vni mapping are mutually exclusive.
- The **no map vlan *vlan\_id*** command removes the VNI mappings for one or more VLANs.
- You cannot delete a VLAN (by means of the **no interface vlan** command) that is referenced by means of the **map vlan vni** command.
- This command does not trigger VLAN provisioning, unlike the behavior of the **attach vlan** command.

## Examples

To configure a manual mapping of VLANs to VNIs in "gateway1":

```
switch(config)# overlay-gateway gateway1
switch(config-overlay-gw-gateway1)# map vlan 10,20-22 vni 5000-5002,6000
```

This results in the following in the running configuration:

```
overlay-gateway gateway1
  type layer2-extension mode vxlan-ipv4
  map vlan 10 vni 5000
  map vlan 20 vni 5001
  map vlan 21 vni 5002
  map vlan 22 vni 6000
```

To configure an automatic mapping of VLANs to VNIs in "gateway1":

```
switch(config)# overlay-gateway gateway1
switch(config-overlay-gw-gateway1)# map vlan vni auto
```

# maps

Activates MAPS configuration mode for all Monitoring and Alerting Policy Suite (MAPS) commands.

## Syntax

**maps**

## Modes

RBridge configuration mode

## Usage Guidelines

This command can be used in conjunction with the `enable`, `email`, `relay`, and `domain-name` commands as a shortcut.

In order to disable MAPS, the **no maps enable** command triggers an HA failover.

## Examples

Typical command example.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# maps
device(config-rbridge-id-5-maps)#
```

## History

Release version	Command history
6.0.1	This command was introduced.

# maps reapply-policy

Reapplies the currently configured MAPS policy.

## Syntax

```
maps reapply-policy [ rbridge-id number]
```

## Parameters

**rbridge-id** *number*

The number of an RBridge node.

## Modes

Privileged EXEC mode.

## Usage Guidelines

If you modify the MAPS policy configuration on a cluster, this command reapplies the policy to the cluster with the updated configuration.

The **rbridge-id** operand reapplies the MAPS policy on a remotely cluster.

## Examples

Typical command example for reapplies the MAPS policy on a remote cluster.

```
device# maps reapply-policy rbridge-id 56
```

## History

Release version	Command history
7.0.1	This command was introduced.

# match

Creates a class map to classify traffic based on configured match criteria.

## Syntax

`match criteria`

## Command Default

The only available match criteria at this time is "match any."

## Parameters

*criteria*

Used while in config-classmap mode to configure the match criteria for the class.

## Modes

Class-map configuration mode

## Usage Guidelines

Use this command to classify traffic based on match criteria. When you launch the **class-map** command, the system is placed in config-classmap mode for the configured map. At this point, you can provide match criteria for the class. The only available match criteria at this time is "match any."

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

## Examples

To configure "match any" match criteria for the class while in config-classmap mode:

```
device(config-classmap)# match any
```

## Related Commands

[class-map](#), [show running-config class-map](#)

# match (route map)

Defines a variety of match conditions for a route map.

## Syntax

`match as-path name`

`match community name exact-match ]`

`match extcommunity number`

`match interface { <N>gigabitethernet rbridge-id / slot / port | loopback number | port-channel number | ve vlan_id }`

`match ip address { acl name [ prefix-list string ] | prefix-list string [ acl name ] }`

`match ip next-hop prefix-list string`

`match ip route-source prefix-list string`

`match ipv6 next-hop prefix-list string`

`match ipv6 route-source prefix-list string`

`match metric num`

`match protocol bgp { external | internal | static-network }`

`match protocol static`

`match route-type { internal | type-1 | type-2 }`

`match tag num`

`match vrf name`

`no match as-path`

`no match community`

`no match extcommunity`

`no match interface { <N>gigabitethernet rbridge-id / slot / port | loopback number | port-channel number | ve vlan_id }`

`no match ip address`

`no match ip next-hop`

`no match ip route-source`

`no match ipv6 address`

`no match ipv6 next-hop`

`no match ipv6 route-source`

`no match metric`

`no match protocol`

`no match route-type`

`no match tag`

## Command Default

This option is disabled.

## Parameters

### *as-path*

Matches an AS-path access list name in a route-map instance.

#### *name*

Name of an AS-path access list. Range is from 1 through 32 ASCII characters.

### **community**

Matches a BGP community access list name in a route-map instance.

#### *name*

Name of a BGP community access list. Values range from 1 through 32 ASCII characters.

### **exact-match**

Matches a route only if the route community attributes field contains the same community numbers specified in the **match** statement.

### **extcommunity** *number*

Matches a BGP extended community list in a route-map instance and specifies an extended community list number. Valid values range from 1 through 99.

### **interface**

Matches interface conditions in a route-map instance.

#### **<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### **loopback** *number*

Specifies a loopback port number. The range is from 1 through 255.

### **port-channel** *number*

Specifies a port-channel interface. The range is from 1 through 6144.

### **ve** *vlan\_id*

Specifies the VLAN number. (Refer to the Usage Guidelines.) The range is from 1 through 8191.

### **ip address**

Matches an IP address in a route-map instance.

**acl** *name*

Name of the access list. Range is from 1 through 32 ASCII characters.

**prefix-list** *string*

Specifies an IP prefix list. Range is from 1 through 32 ASCII characters.

**ip next-hop**

Matches IP next-hop match conditions in a route-map instance.

**ip route-source**

Matches an IP route source in a route-map instance.

**ipv6 address**

Matches an IPv6 address in a route-map instance.

**ipv6 next-hop**

Matches IPv6 next-hop match conditions in a route-map instance.

**ipv6 route-source**

Matches an IPv6 route source in a route-map instance.

**metric** *num*

Matches a route metric in a route-map instance. Values range from 0 through 4294967295.

**protocol bgp external**

Matches on BGP routes.

**protocol bgp internal**

Matches on iBGP routes.

**protocol bgp static-network**

Matches on BGP4 static network routes. This is applicable only for BGP outbound policy.

**protocol static**

Matches on static routes.

**route-type**

Matches a route type in a route-map instance.

**internal**

Internal route type

**type-1**

OSPF external route type 1

**type-2**

OSPF external route type 2

**tag** *tag-value*

Specifies a route tag and route tag value.

**vrf** *name*

Specifies a non-default VRF. Valid values range from 0 through 4294967295.

## Modes

Route-map configuration mode



## Usage Guidelines

The **no** form of the command restores the default.

## Examples

The following example matches AS-path ACL 1 in route-map instance "myroutes".

```
device#configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes permit 10
device(config-route-map myroutes/permit/10)# match as-path 1
```

## Related Commands

[ip as-path access-list](#), [route-map](#)

# match access-list

Configures the access control list to be used with the class map for flow-based QoS.

## Syntax

```
match access-list acl_name
```

## Parameters

*acl\_name*

Any valid Layer 2 or Layer 3 ACL access list name.

## Modes

Class-map configuration mode

## Examples

Example command:

```
switch(config-classmap)# match access-list engineeringACL
```

## Related Commands

[class-map](#)

# match as-path

Matches an AS-path access list name in a route-map instance.

## Syntax

```
match as-path name
```

```
no match as-path
```

## Parameters

*name*

Name of an AS-path access list. Range is from 1 through 32 ASCII characters.

## Modes

Route-map configuration mode

## Related Commands

[route-map](#)

# match community

Matches a BGP community access list name in a route-map instance.

## Syntax

`match community name`

`no match community`

## Parameters

*name*

Name of a BGP community access list. Values range from 1 through 32 ASCII characters.

## Modes

Route-map configuration mode

## Usage Guidelines

Enter `no match community name` to disable this feature.

## Examples

The following example matches the gshut community attribute in a route map instance.

```
device# config terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# route-map myroutes permit 10
device(config-route-map-myroutes/permit/10)# match community gshut
```

## History

Release version	Command history
7.2.0	This command was modified to include support for matching the gshut community attribute.

## Related Commands

[route-map](#)

# match extcommunity

Matches a BGP extended community list in a route-map instance.

## Syntax

**match extcommunity** *number*

**no match extcommunity**

## Command Default

BGP extended community access list names are not matched.

## Parameters

*name*

Extended community list number. Values range from 1 through 99.

## Modes

Route-map configuration mode.

## Usage Guidelines

Enter **no match extcommunity** to remove the community match statement from the configuration file.

## Examples

To configure a route map that matches on extended community ACL 1.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip extcommunity-list 1 permit 123:2
device(config-rbridge-id-122)# route-map extComRmap permit 10
device(config-route-map-extComRmap/permit/10)# match extcommunity 1
```

## History

Release version	Command history
5.0.0	This command was introduced.

# match interface

Matches interface conditions in a route-map instance.

## Syntax

```
match interface { <N>gigabitethernet rbridge-id / slot / port | loopback number | port-channel number | ve vlan_id }
no match interface
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### loopback *number*

Specifies a loopback port number. The range is from 1 through 255.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

### ve *vlan\_id*

Specifies the VLAN number. (Refer to the Usage Guidelines.) The range is from 1 through 8191.

## Modes

Route-map configuration mode

## Usage Guidelines

Use this command to configure the interface match clause in a route-map instance. A maximum of three interfaces is supported.

## Examples

The following example configures a route-map that matches on an interface.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# route-map myintroutemap1 permit 99
device(config-rbridge-id-5)# match interface ten 5/1/1 ten 5/3/2
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

## Related Commands

[route-map](#)

# match ip address

Matches IP address conditions in a route-map instance.

## Syntax

```
match ip address acl name
```

```
no match ip address acl name
```

## Parameters

*acl name*

Name of the access list. Range is from 1 through 32 ASCII characters.

## Modes

Route-map configuration mode

## Usage Guidelines

Use this command to specify an IP prefix match clause in a route-map instance.



# match ip next-hop

Matches IP next-hop match conditions in a route-map instance.

## Syntax

```
match ip next-hop prefix-list name
```

```
no match ip next-hop
```

## Parameters

**prefix-list** *name*

Specifies a prefix list. Values range from 1 through 32 ASCII characters.

## Modes

Route-map configuration mode

## Usage Guidelines

Use this command to specify an IP next-hop match clause in a route-map instance.

# match ipv6 address

Matches IPv6 address conditions in a route map instance.

## Syntax

```
match ipv6 address [ prefix-list prefix-list-name ]
no match ipv6 address
```

## Command Default

No routes are distributed based on destination network number.

## Parameters

**prefix-list** *prefix-list-name*  
Specifies the name of an IPv6 prefix list.

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the **match ipv6 address** entry.

## Examples

This example matches IPv6 routes that have addresses specified by the prefix list named "myprefixlist".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map extComRmap permit 10
device(config-route-map-sendExtComRmap/permit/10)# match ipv6 address prefix-list myprefixlist
```

## History

Release version	Command history
5.0.0	This command was introduced.

# match metric

Matches a route metric in a route-map instance.

## Syntax

`match metric value`

`no match metric`

## Parameters

*value*

Route metric. Values range from 0 through 4294967295.

## Modes

Route-map configuration mode

## Usage Guidelines

Use this command to specify a route-map metric in route-map instance.

# match protocol bgp

Matches BGP routes on protocol types and subtypes in a route-map instance.

## Syntax

```
match protocol bgp [ external | internal | static-network ]
```

```
no match protocol bgp
```

## Parameters

### external

Matches EBGP routes.

### internal

Matches IBGP routes.

### static-network

Matches BGP static routes. This is applicable only for BGP outbound policy.

## Modes

Route-map configuration mode

## Related Commands

[route-map](#)

# match route-type

Matches a route type in a route-map instance.

## Syntax

```
match route-type [ internal | type-1 | type-2 ]
```

```
no match route-type
```

## Parameters

**internal**

Internal route type

**type-1**

OSPF external route type 1

**type-2**

OSPF external route type 2

## Modes

Route-map configuration mode

## Related Commands

[route-map](#)

# match tag

Matches a route tag in a route-map instance.

## Syntax

`match tag value`

`no match tag`

## Parameters

*value*

The range of valid values is from 0 through 4294967295.

## Modes

Route-map configuration mode

## Related Commands

[route-map](#)

# max-age

Sets the interval time in seconds between messages that the spanning tree receives from the interface.

## Syntax

**max-age** *seconds*

**no max-age**

## Command Default

20 seconds.

## Parameters

*seconds*

Configures the STP interface maximum age. Valid values range from 6 through 40.

## Modes

Spanning tree configuration mode

## Usage Guidelines

Use this command to control the maximum length of time that passes before an interface saves its configuration Bridge Protocol Data Unit (BPDU) information.

If the **vlan** parameter is not provided, the *seconds* value is applied globally for all per-VLAN instances. However, for VLANs that have been configured explicitly, the per-VLAN configuration takes precedence over the global configuration.

When configuring the maximum age, the **max-age** command setting must be greater than the **hello-time** command setting. The following relationship should be kept:

$$(2 \times (\text{forward-delay} - 1)) \geq \text{max-age} \geq (2 \times (\text{hello-time} + 1))$$

Enter **no max-age** to return to the default configuration.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes.

## Examples

To configure the maximum age to 10 seconds:

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# max-age 10
```

```
device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# max-age 10
```

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# max-age 10
```

```
device# configure terminal
device(config)# protocol spanning-tree pvst
device(conf-pvst)# max-age 10
```

```
device# configure terminal
device(config)# protocol spanning-tree rpvt
device(conf-rpvt)# max-age 10
```

## Related Commands

[forward-delay](#), [hello-time](#)



# max-hops

Configures the maximum number of hops for a Bridge Protocol Data Unit (BPDU) in an MSTP region.

## Syntax

```
max-hops hop_count  
no max-hops
```

## Command Default

20 hops

## Parameters

*hop\_count*

Specifies the maximum number of hops for which the BPDU will be valid. Valid values range from 1 through 40.

## Modes

Protocol Spanning Tree MSTP configuration mode

## Usage Guidelines

Specifying the maximum hops for a BPDU prevents the messages from looping indefinitely on the interface. When you change the number of hops, it affects all spanning-tree instances.

Enter **no max-hops** to return to the default value.

## Examples

To set the number of maximum hops to 25 for all MSTPs:

```
switch(config)# protocol spanning-tree mstp  
switch(conf-mstp)# max-hops 25
```

## Related Commands

[show spanning-tree mst brief](#)

# max-mcache

Configures the maximum multicast cache size.

## Syntax

```
max-mcache num  
no max-mcache
```

## Command Default

Multicast cache size is 24576 entries.

## Parameters

*num*  
Number of entries in the multicast cache. Valid values range from 1 through 24576.

## Modes

Router PIM configuration mode

## Usage Guidelines

Entering the **no** form of the command sets the maximum multicast cache size to the default - 24576 entries.

## Examples

Setting the multicast cache to 500 entries.

```
device(config)# router pim  
device(conf-pim-router)# max-mcache 500
```

# max-metric router-lsa

Advertises the maximum metric value in different Link State Advertisements (LSAs).

## Syntax

```
max-metric router-lsa [ all-vrfs ] [ all-lsas | external-lsa metric-value | link { all | ptp | stub | transit } | summary-lsa metric-value | on-startup { time | wait-for-bgp [ all-lsas | summary-lsa metric-value | external-lsa metric-value | link { all | ptp | stub | transit } ] }
```

```
no max-metric router-lsa [ all-vrfs ] [ all-lsas | external-lsa | link { all | ptp | stub | transit } | summary-lsa | on-startup { time | wait-for-bgp [ all-lsas | link { all } ] }
```

## Parameters

### all-vrfs

Applies the configuration change to all instances of OSPF.

### all-lsas

Sets the **summary-lsa** and **external-lsa** optional parameters to the corresponding default max-metric value. For a non-default instance of OSPF, only the summary-lsa and external-lsa parameters are set.

### external-lsa *metric-value*

Modifies the metric of all external type 5 LSAs to equal the specified value or a default value. The range for metric value is 1 to 16777214 (0x00001 - 0x00FFFFE), and the default is 16711680 (0x00FF0000).

### link

Specifies the types of links for which the maximum metric is advertised. By default, the maximum metric is advertised only for transit links.

#### all

Advertises the maximum metric in Router LSAs for all supported link types.

#### ptp

Advertises the maximum metric in Router LSAs for point-to-point links.

#### stub

Advertises the maximum metric in Router LSAs for stub links.

#### transit

Advertises the maximum metric in Router LSAs for transit links. This is the default link type.

### summary-lsa *metric-value*

Modifies the metric of all summary type 3 and type 4 LSAs to equal the specified value or a default value. The range for metric value is 1 to 16777215 (0x00001 - 0x00FFFFE), and the default is 16711680 (0x00FF0000).

### on-startup

Applies the configuration change at the next OSPF startup.

#### *time*

Sets the time (in seconds) for which the specified links in Router LSAs are advertised when the metric is set to the maximum value of 0xFFFF. The range for *time* is 5 to 86,400.

**wait-for-bgp**

Indicates that OSPF should wait for either 600 seconds or until BGP has finished route table convergence, whichever happens first, before advertising the links with the normal metric.

## Modes

OSPF router configuration mode

OSPF VRF router configuration mode

## Usage Guidelines

Use this command to set the maximum metric value advertised in different Link State Advertisements (LSAs). When enabled, the router configures the maximum value of the metric for routes and links advertised in various types of LSAs. Because the route metric is set to its maximum value, neighbors will not route traffic through this router except to directly connected networks. Thus, the device becomes a stub router, which is desirable when you want:

- Graceful removal of the router from the network for maintenance.
- Graceful introduction of a new router into the network.
- To avoid forwarding traffic through a router that is in critical condition.

Enter **no max-metric router-lsa all-lsas** to disable advertising the maximum metric value in different LSAs.

## Examples

The following example advertises the maximum metric value using the **all-lsas** option.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# max-metric router-lsa all-lsas
```

The following example configures an OSPF device to advertise a maximum metric for 72 seconds after a restart before advertising with a normal metric for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf vrf green
device(config-router-ospf-vrf-red)# max-metric router-lsa on-startup 72
```

# max-metric router-lsa (OSPFv3)

Advertises the maximum metric value in different Link State Advertisements (LSAs).

## Syntax

```
max-metric router-lsa [ all-lsas | external-lsa metric-value | include-stub | on-startup { time | wait-for-bgp } | summary-lsa metric-value ]
```

```
no max-metric router-lsa [ all-lsas | external-lsa | include-stub | on-startup { time | wait-for-bgp } | summary-lsa ]
```

## Parameters

### all-lsas

Sets the **summary-lsa** and **external-lsa** optional parameters to the corresponding default max-metric value. For a non-default instance of OSPFv3, only the summary-lsa and external-lsa parameters are set.

### external-lsa *metric-value*

Configures the maximum metric value for all external type-5 and type-7 LSAs. The range for metric value is 1 to 16777214 (0x00001 - 0x00FFFFFFE), and the default is 16711680 (0x00FF0000).

### include-stub

Specifies the advertisement of the maximum metric value for point-to-point and broadcast stub links in the intra-area-prefix LSA..

### on-startup

Applies the configuration change at the next OSPF startup.

#### *time*

Sets the time (in seconds) for which the specified links in Router LSAs are advertised when the metric is set to the maximum value of 0xFFFF. The range for *time* is 5 to 86400.

### wait-for-bgp

Specifies that OSPFv3 should wait until BGP has finished route table convergence before advertising the links with the normal metric, or for no more than 600 seconds.

### summary-lsa *metric-value*

Configures the maximum metric value for all summary type 3 and type 4 LSAs. The range for metric value is 1 to 16777215 (0x00001 - 0x00FFFFFFE), and the default is 16711680 (0x00FF0000).

## Modes

OSPFv3 router configuration mode

OSPFv3 VRF router configuration mode

## Usage Guidelines

Use this command to set the maximum metric value advertised in different Link State Advertisements (LSAs). When enabled, the router configures the maximum value of the metric for routes and links advertised in various types of LSAs. Because the route metric is set to its maximum value, neighbors will not route traffic through this router except to directly connected networks. Thus, the device becomes a stub router, which is desirable when you want:

- Graceful removal of the router from the network for maintenance.
- Graceful introduction of a new router into the network.
- To avoid forwarding traffic through a router that is in critical condition.

Enter **no max-metric router-lsa** to disable advertising the maximum metric value in different LSAs.

## Examples

The following example configures an OSPFv3 device to advertise a maximum metric and sets the maximum metric value for all external type-5 and type-7 LSAs to 1000.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# max-metric router-lsa external-lsa 1000
```

The following example configures an OSPFv3 device to advertise a maximum metric and specifies the advertisement of the maximum metric value for point-to-point and broadcast stub links in the intra-area-prefix LSA.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# max-metric router-lsa include-stub
```

The following example configures an OSPFv3 device to advertise a maximum metric for 75 seconds after a restart before advertising with a normal metric for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf vrf red
device(config-ipv6-router-ospf-vrf-red)# max-metric router-lsa on-startup 75
```

The following example configures an OSPFv3 device to advertise a maximum metric until BGP routing tables converge or until the default timer of 600 seconds expires.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# max-metric router-lsa on-startup wait-for-bgp
```

The following example configures an OSPFv3 device to advertise a maximum metric and sets the maximum metric value for all summary type-3 and type-4 LSAs to 100.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# max-metric router-lsa summary-lsa 100
```

## History

Release version	Command history
6.0.1	This command was introduced.

# max-route

Specifies the maximum number of routes allowed in the routing table per VRF instance, for an IPv4 or IPv6 VRF address family.

## Syntax

**max-route** *value*

## Command Default

If this command is not configured, the maximum allowed number of routes, 4294967295 (see Parameters), is applied. This number does not appear in a running configuration.

## Parameters

*value*

The maximum allowed number of routes. Range is from 1 through 4294967295.

## Modes

VRF address-family IPv4 and IPv6 configuration modes

## Examples

To configure the maximum number of allowed routes to 3600 for VRF "myvrf" for an IPv4 address family:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf myvrf
device(config-vrf-myvrf)# address-family ipv4 unicast
device(vrf-myvrf-ipv4-unicast)# max-route 3600
```

## History

Release version	Command history
7.0.1	This command was modified to include an example.



# maxas-limit

Imposes a limit on the number of autonomous systems in the AS-PATH attribute.

## Syntax

```
maxas-limit in num
no maxas-limit { in }
```

## Command Default

This option is disabled.

## Parameters

**in**

Allows an AS-PATH attribute from any neighbor to impose a limit on the number of autonomous systems.

*num*

Range is from 0 through 300. The default is 300.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

BGP address-family IPv6 unicast VRF configuration mode is not supported. The IPv6 configuration is inherited from an IPv4 configuration.

## Examples

To set the limit on the number of BGP4 autonomous systems in the AS-PATH attribute to 100 on an RBridge (for the default VRF).

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# maxas-limit in 100
```

To set the limit on the number of autonomous systems in the AS-PATH attribute to 100 for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# maxas-limit in 100
```

## History

Release version	Command history
6.0.1	Support was added for the BGP address-family IPv4 unicast VRF configuration modes.

# maximum-paths (BGP)

Sets the maximum number of BGP4 and BGP4+ shared paths.

## Syntax

```
maximum-paths num | use-load-sharing
no maximum-paths
```

## Command Default

This option is disabled.

## Parameters

*num*

Maximum number of paths across which the device balances traffic to a given BGP4 destination. The range is from 1 through 32 for the Extreme VDX 8770 and Extreme VDX 6940; the range is from 1 through 16 for the Extreme VDX 6740. The default is 1 for all platforms.

**use-load-sharing**

Uses the maximum IP ECMP path value.

## Modes

BGP address-family IPv4 unicast configuration mode  
 BGP address-family IPv6 unicast configuration mode  
 BGP address-family IPv4 unicast VRF configuration mode  
 BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

Use this command to change the maximum number of BGP4 shared paths.

## Examples

This example sets the maximum number of BGP4 shared paths to 8.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# maximum-paths 8
```

This example sets the maximum number of BGP4+ shared paths.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# maximum-paths use-load-sharing
```

This example sets the maximum number of BGP shared paths to 2 in a nondefault VRF instance in the IPv6 address family.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# maximum-paths 2
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

## Related Commands

[maximum-paths ebgp ibgp](#)

# maximum-paths (OSPF)

Changes the maximum number of OSPF shared paths.

## Syntax

**maximum-paths** *num*

**no maximum-paths**

## Parameters

*num*

Maximum number of paths across which the device balances traffic to a given OSPF destination. The range is from 1 through 32. The default is 8.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the default.

## Examples

The following example sets the maximum number of shared paths to 22.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router ospf
device(config-router-ospf-vrf-default-vrf)# maximum-paths 22
```

The following example sets the maximum number of shared paths to 25 for VRF "green".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router ospf vrf green
device(config-router-ospf-vrf-vrf-green)# maximum-paths 25
```

## History

Release version	Command history
6.0.1	This command was introduced.

# maximum-paths ebgp ibgp

Specifies the number of equal-cost multipath eBGP or iBGP routes or paths that are selected.

## Syntax

```
maximum-paths { ebgp num | ibgp num }  
no maximum-paths
```

## Command Default

This option is disabled.

## Parameters

### ebgp

Specifies eBGP routes or paths.

### ibgp

Specifies iBGP routes or paths.

### *num*

The number of equal-cost multipath routes or paths that are selected. Range is from 1 through 32. 1 disables equal-cost multipath.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

Enhancements to BGP load sharing support the load sharing of BGP4 and BGP4+ routes in IP Equal-Cost Multipath (ECMP), even if the BGP multipath load-sharing feature is not enabled by means of the **use-load-sharing** option to the **maximum-paths** command. You can set separate values for IGMP and ECMP load sharing. Use this command to specify the number of equal-cost multipath eBGP or iBGP routes or paths that are selected.

## Examples

This example sets the number of equal-cost multipath eBGP routes or paths that will be selected to 6 in the IPv4 address family.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# maximum-paths ebgp 6
```

This example sets the number of equal-cost multipath iBGP routes or paths that will be selected to 4 in the IPv6 address family.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# maximum-paths ibgp 4
```

This example sets the number of equal-cost multipath eBGP routes or paths that will be selected to 3 for IPv4 VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# maximum-paths ebgp 3
```

This example sets the number of equal-cost multipath iBGP routes or paths that will be selected to 2 for IPv6 VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# maximum-paths ibgp 2
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# med-missing-as-worst

Configures the device to favor a route that has a Multi-Exit Discriminator (MED) over a route that does not have one.

## Syntax

```
med-missing-as-worst
no med-missing-as-worst
```

## Modes

BGP configuration mode

## Usage Guidelines

When MEDs are compared, by default the device favors a low MED over a higher one. Because the device assigns a value of 0 to a route path MED if the MED value is missing, the default MED comparison results in the device favoring the route paths that do not have MEDs.

The **no** form of the command restores the default where a device does not favor a route that has a MED over other routes.

## Examples

The following example configures the device to favor a route containing a MED.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# med-missing-as-worst
```



# message-interval

Configures the Protocol Independent Multicast (PIM) Join or Prune message interval.

## Syntax

```
message-interval num  
no message-interval num
```

## Command Default

60 seconds

## Parameters

*num*

The interval value in seconds. Valid values range from 10 through 65535 seconds.

## Modes

Router PIM configuration mode

## Usage Guidelines

Use this command to specify the interval at which the periodic PIM Join or Prune messages must be sent out.

Enter the **no** form of the command to disable this feature.

## Examples

Setting the interval to one hour.

```
device(config)# router pim  
device(conf-pim-router)# message-interval 3600
```

## Related Commands

[router pim](#)

# metric-type

Configures the default metric type for external routes.

## Syntax

```
metric-type { type1 | type2 }  
no metric-type { type1 | type2 }
```

## Command Default

Type 1

## Parameters

### type1

The metric of a neighbor is the cost between itself and the device plus the cost of using this device for routing to the rest of the world.

### type2

The metric of a neighbor is the total cost from the redistributing device to the rest of the world.

## Modes

- OSPF router configuration mode
- OSPFv3 router configuration mode
- OSPF router VRF configuration mode
- OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the default setting. You must specify a type parameter when using the **no** form.

## Examples

The following example sets the default metric type for external routes to type 2.

```
device# configure terminal  
device(config)# rbridge-id 5  
device(config-rbridge-id-5)# router ospf  
device(config-router-ospf-vrf-default-vrf)# metric-type type2
```

## History

Release version	Command history
5.0.0	Support was added for OSPFv3.

## Related Commands

[default-information-originate \(OSPFv2\)](#)

# minimum-links

Configures the minimum bandwidth or number of links to be running to allow the port-channel to function.

## Syntax

```
minimum-links num-of-links  
no minimum-links
```

## Command Default

Number of links is 1.

## Parameters

*num-of-links*  
The number of links. Valid values range from 1 through 32.

## Modes

Port-channel interface configuration mode

## Usage Guidelines

Use this command to allow a port-channel to operate at a certain minimum bandwidth all the time. If the bandwidth of the port-channel drops below that minimum number, then the port-channel is declared operationally DOWN even though it has operationally UP members.

Enter **no minimum-links** to restore the default value.

## Examples

The following example sets the minimum number of links to 16 on a specific port-channel interface.

```
switch(config)# interface port-channel 33  
switch(config-Port-channel-33)# minimum-links 16
```

# mode (LLDP)

Sets the LLDP mode on the device.

## Syntax

```
mode { tx | rx }
```

## Command Default

Both transmit and receive modes are enabled.

## Parameters

- tx**  
Specifies to enable only the transmit mode.
- rx**  
Specifies to enable only the receive mode.

## Modes

Protocol LLDP configuration mode

## Examples

To enable only the transmit mode:

```
device(conf-lldp)# mode tx
```

To enable only the receive mode:

```
device(conf-lldp)# mode rx
```

## Related Commands

[show lldp interface](#)

# mode (27x40 GbE line card)

Sets Performance or Density operating modes on the 27x40 GbE line card when you are logged into a VDX 8770.

## Syntax

`mode performance`

`no mode performance`

## Command Default

Density mode (`no mode performance`) is enabled.

## Modes

port-group

## Usage Guidelines

Use this command to set Performance or Density (default) operating modes for port groups 1-9 on the 27x40 GbE line card. When a port group is configured in Performance mode, the third port in the port group is persistently disabled, but the remaining two ports operate at up to 40 Gbps in Performance mode to achieve the 80 Gbps maximum rate for the port group. QSFP breakout mode is only supported on ports configured in Performance mode.

If Density mode (default) is configured for a port group, all three ports in the group are enabled in Density mode, so cannot support the 40 Gbps maximum rate. If this mode is configured on all port groups, 27 total ports are available for use.

For more information on port groups and the Performance and Density operating modes on the line card, refer to the "Overview" chapter in the Extreme VDX 8770-4 and VDX 8770-8 Hardware Reference Manuals.

Enter `no performance mode` to restore the default value. Power off the line card before configuring operating modes.

## Examples

To enable Performance mode on port group 9 of the line card in slot 3 of switch with RBridge ID 1.

```
switch# configure terminal
Entering configuration mode terminal
switch(config)# hardware
switch(config-hardware)# port-group 1/3/9
switch(config-port-group-1/3/9)# mode performance
%Warning: port-group mode performance is a disruptive command.
Please save the running-config to startup-config and a power-cycle for the
changes to take place.
```

## Related Commands

[hardware](#), [port-group](#)

# mode (VDX6940-144S)

Enables 100 GbE or 40 GbE operation on VDX 6940-144S dual personality ports.

## Syntax

```
mode 100g
```

```
mode 40g
```

## Command Default

(**mode 40g**) is enabled for all dual personality port groups.

## Modes

port-group

## Usage Guidelines

Use this command to set the operation of dual personality port groups on the VDX 6940-144S to either 100 GbE or 40 GbE. A dual personality port group consists of a row of 40 GbE ports located on the right side of the VDX 6940-144S front panel. There are four rows of ports, each row containing three ports. Although only the left-most port in each row operates as a dual personality port (100 GbE or 40 GbE), you configure the entire row for either 100 GbE or 40 GbE operation using the **mode** command. After enabling 100 GbE mode for a port group, the dual personality port operates at 100 GbE, while the remaining two ports in the row are disabled. After you enable 40 GbE mode for a port group, all ports in the row can operate at 40 GbE.

The following table shows the port numbers belonging to each port group, 1 through 4.

### NOTE

**TABLE 10** Dual personality port groups

Port Group	40/100 GbE port # (dual personality port)	40 GbE port #	40 GbE port #
1	97	99	101
2	98	100	102
3	103	105	107
4	104	106	108

Before using this command, perform the following tasks:

- Install a 40 GbE transceiver in the dual personality port for 40 GbE operation or a 100 GbE transceiver in the port for 100 GbE operation.
- Disable all interfaces in the port group using the **shutdown** command. The following example illustrates how to disable port 97 in port group 1. You must also disable ports 99 and 101 in that port group.

```
switch# configure terminal
Entering configuration mode terminal
switch(config)# interface fortygigabitethernet 1/0/97
switch(config-if-fo-1/0/97)# shutdown
```

After using the mode command to enable 100 GbE or 40 GbE operation, you must reboot the switch to enable the set mode.

For more information on VDX 6940-144S ports and the Dual Personality Ports feature, refer to the *ExtremeSwitching VDX 6940 Hardware Installation Guide*.

## Examples

The following example enables 100 GbE operation for port group 1 through 4. If an appropriate 100 GbE transceiver is installed in port 97, this port will operate at 100 GbE after switch reboot and ports 99 and 100 will be disabled.

### NOTE

The **port-group** command uses variable *rbridge-id/slot/port-group-id* to identify the port. To configure a dual personality port group, use 0 for the slot and 1 through 4 for the port-group-id. Slot 0 is always used for a fixed switch, such as the VDX 6940-144S.

```
switch# configure terminal
Entering configuration mode terminal
switch(config)# hardware
switch(config-hardware)# port-group 1/0/1
switch(config-port-group-1/0/1)# mode 100g
2015/12/16-15:27:07, [NSM-1010], 13849, SW/0 | Active | DCE, INFO, sw0, InterfaceMode
changed from None to L2 for interface FortyGigabitEthernet 1/0/97.
%Warning: port-group mode change is a disruptive command.
Please do linecard power-on or switch reload for the changes to take place.
```

The following example enables 40 GbE operation for port group 1. If an appropriate 40 GbE transceiver is installed in port 97, this port will operate at 40 GbE after switch reboot and all ports in the group will operate at 40 GbE.

### NOTE

The **port-group** command uses variable *rbridge-id/slot/port-group-id* to identify the port. To configure a dual personality port group, use 0 for the slot and 1 through 4 for the port-group-id. Slot 0 is always used for a fixed switch, such as the VDX 6940-144S.

```
switch# configure terminal
Entering configuration mode terminal
switch(config)# hardware
switch(config-hardware)# port-group 1/0/1
switch(config-port-group-1/0/1)# mode 40g
2015/12/16-15:27:07, [NSM-1010], 13849, SW/0 | Active | DCE, INFO, sw0, InterfaceMode
changed from L2 to None for interface HundredGigabitEthernet 47/0/98.
%Warning: port-group mode change is a disruptive command.
Please do linecard power-on or switch reload for the changes to take place.
```



# modes

Enables and disables operating modes for port groups for Access Gateway mode.

## Syntax

**modes** *mode\_name*

**no modes** *mode\_name*

## Command Default

**lb**

## Parameters

*mode\_name*

**lb** (Automatic Login Balancing)

## Modes

Port Grouping configuration mode

## Usage Guidelines

Login Balancing (LB) is the only mode that you can enable. Automatic Login Balancing is enabled by default for a port group when the port group is created. If LB mode is enabled for a port group and a VF\_Port goes offline, logins in the port group are redistributed among the remaining VF\_Ports. Similarly, if an N\_Port comes online, port logins in the port group are redistributed to maintain a balanced N\_Port-to-VF\_Port ratio.

You must be in Port Grouping configuration mode for a specific port group to use this command. Entering **no modes mode\_name** disables the mode.

Consider the following when using LB mode with **show running-config ag** and **show ag** commands:

- The only Port Grouping mode that you can enable or disable is LB mode.
- When LB mode is disabled in a port group, the **show running-config ag**, **show ag map**, and **show ag** commands display the configured VF\_Port to N\_Port mapping. This is because configured and active mapping are the same.
- When LB mode is enabled in a port group, **show ag**, and **show ag map** displays the active mapping only because VF\_Port to N\_Port mapping is based on the current distributed load across all N\_Ports. The **show running-config ag** command displays the configured mapping only.

## Examples

Enable Automatic Login Balancing mode on port group 8.

```
sw0(config-rbridge-id-3-ag-pg-8)# modes lb
```

modes

Disable Automatic Login Balancing mode on port group 8.

```
sw0(config-rbridge-id-3-ag-pg-8) # no modes lb
```

# monitor session

Enables a Port Mirroring session for monitoring traffic.

## Syntax

```
monitor session session_number
```

```
no monitor session session_number
```

## Parameters

*session\_number*

Specifies a session identification number. Valid values range from 1 through 512.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no monitor session** to delete the port mirroring session.

## Examples

To enable session 22 for monitoring traffic:

```
switch# configure  
switch(config)# monitor session 22
```

## Related Commands

[source](#)

# monitor session (VXLAN)

Enables switched port analyzer (SPAN) on one or all tunnels of a VXLAN overlay gateway.

## Syntax

```
monitor session session_number direction { tx | rx | both } [ remote-endpoint { ip_address | any } ] vlan [ add | remove ]
    VLAN_ID_range
```

```
no monitor session session_number
```

## Parameters

*session\_number*

Specifies the SPAN session ID that was configured with the global **monitor session** command.

**tx**

Enables SPAN for the transmitting tunnels.

**rx**

Enables SPAN for the receiving tunnels.

**both**

Enables SPAN for both the transmitting and receiving tunnels.

*ip\_address*

Enables SPAN for the specified the IPv4 address of the remote Hypervisor for the NSX Controller to VXLAN termination endpoint (VTEP).

**any**

Enables SPAN for all tunnels on the gateway.

**add**

Enables SPAN on specified VLAN IDs. You can use this option if you have disabled SPAN on specific VLAN IDs and now want to re-enable SPAN on these IDs.

**remove**

Disables SPAN on specified VLAN IDs.

*VLAN\_ID\_range*

Specifies the VLAN IDs for enabling SPAN.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

Use this command to enable SPAN on one or all tunnels of this gateway for specified VLANs. You can use the **remote-endpoint** option to choose all tunnels or specific tunnels of this gateway. You choose a specific tunnel by specifying the remote Hypervisor 's VTEP IP address. This address is matched with the destination IP address of the tunnels.

The **remove** option can be used to exclude VLANs from a previously configured list. If all the VLANs are removed, the entire SPAN configuration is deleted (this is the same behavior as that resulting from the **no monitor session session\_number** command).

The only way to change the direction once you have run this command is to remove the SPAN configuration, then rerun the **monitor session** command. Specified VLANs must already be configured as exported through this gateway.

The SPAN session number must already be configured, and the SPAN destination must already be specified and cannot be a tunnel.

The SPAN session must not include source port configuration for this gateway.

The deletion of an attached VLAN (by using the **no attach vlan** command) is blocked if SPAN has been enabled for the VLAN you are trying to delete.

The **no** form of this command removes SPAN configuration for the gateway.

## Examples

To enable SPAN for all tunnels in both directions for "gateway1" on VLAN IDs 1 through 10 and SPAN session ID 3:

```
switch# configure
switch(config)# overlay-gateway gateway1
switch(config-overlay-gw-gateway1)# monitor session 3 direction both remote-endpoint any vlan add 1-10
```

## Related Commands

[overlay-gateway](#)

# mtrace

Enables the diagnostic tracing of a multicast path from a specified source to a destination for a multicast group.

## Syntax

```
mtrace [ ipv6 ] [ vrfvrf-name ] { source source-ip-address destination destination-ip-address } [ group group-multicast-address ]
```

```
no mtrace
```

## Command Default

See Parameters for default IPv4 and IPv6 group addresses.

## Parameters

**ipv6**

Specifies tracing in an IPv6 network.

**vrfvrf-name**

Specifies a VRF for tracing in an IPv4 or IPv6 network.

**source***source-ip-address*

Specifies an IPv4 or IPv6 address of a multicast-capable source. This is a unicast address at the beginning of the path to be traced.

**destination***destination-ip-address*

Specifies an IPv4 or IPv6 unicast destination address. If omitted, the trace starts from the system where the command is issued.

**group***group-multicast-address*

Specifies an IPv4 or IPv6 multicast address of a group to be traced. The default IPv4 group address is 225.1.1.1. The default IPv6 group address is FFOE:0:0:0:0:0:10E (IETF-2\_AUDIO).

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **no** form of this command to disable multicast tracing.

## Examples

This IPv6 example uses an optional group multicast address.

```
device# mtrace ipv6 source 102::1 destination 101::2 group ff1d::2

Mtrace handle query from src 102::1 to dest 101::2 through group ff1d::2
Collecting Statistics, waiting time 5 seconds.....
    Type Control-c to abort
0 12::1 PIM thresh^ 1 MTRACE_NO_ERR
1 13::1 PIM thresh^ 1 MTRACE_NO_ERR
2 102::2 PIM thresh^ 1 MTRACE_REACHED_RP
```

## History

Release version	Command history
7.0.0	This command was introduced.

# mtu

Specifies the size of the maximum transmission unit (MTU) on an interface.

## Syntax

**mtu** *size*

**no mtu**

## Command Default

Interfaces have a default MTU of 2500 bytes.

## Parameters

*size*

Size, in bytes, of the MTU. Range is from 1522 through 9216.

## Modes

Global configuration mode

Interface subtype configuration mode

## Usage Guidelines

Configuring an MTU on a VLAN interface is not valid.

If you use the **ipv6 mtu** command to change the MTU value for IPv6 functionality, you must set the same value on the interface by using the **mtu** command. Otherwise packets will be dropped.

The **no mtu** command returns the MTU size to the default value.

The only MTU size available on a VXLAN is 9156 bytes, due to hardware restrictions.

## Examples

The following example sets the MTU to 2000 bytes for every interface.

```
device# configure terminal
device(config)# mtu 2000
```

On a specified Ethernet interface, the following example sets the MTU to 2000 bytes.

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# mtu 2000
```



# multipath

Changes load sharing to apply to only iBGP or eBGP paths, or to support load sharing among paths from different neighboring autonomous systems.

## Syntax

```
multipath { ebgp | ibgp | multi-as }
no multipath { ebgp | ibgp | multi-as }
```

## Command Default

This option is disabled.

## Parameters

- ebgp**  
Enables load sharing of eBGP paths only.
- ibgp**  
Enables load sharing of iBGP paths only.
- multi-as**  
Enables load sharing of paths from different neighboring autonomous systems.

## Modes

- BGP address-family IPv4 unicast configuration mode
- BGP address-family IPv6 unicast configuration mode
- BGP address-family IPv4 unicast VRF configuration mode
- BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

By default, when BGP load sharing is enabled, both iBGP and eBGP paths are eligible for load sharing, while paths from different neighboring autonomous systems are not.

## Examples

This example changes load sharing to apply to iBGP paths in the IPv4 address family.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# multipath ibgp
```

This example enables load sharing of paths from different neighboring autonomous systems in the IPv6 address family.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# multipath multi-as
```

This example changes load sharing to apply to eBGP paths in IPv4 VRF instance "red":

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# multipath ebgp
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# multiplier (LLDP)

Sets the number of consecutive misses of hello messages before LLDP declares the neighbor as dead.

## Syntax

`multiplier value`

`no multiplier`

## Command Default

Multiplier default value is 4.

## Parameters

*value*

Specifies a multiplier value to use. Valid values range from 2 through 10.

## Modes

Protocol LLDP and profile configuration modes

## Usage Guidelines

Enter `no multiplier` to return to the default setting.

The LLDP multiplier can also be configured for a specific LLDP profile. When you apply an LLDP profile on an interface using the `lldp profile` command, it overrides the global configuration. If a profile is not present, then the default global profile is used until you create a valid profile.

## Examples

To set the number of consecutive misses:

```
device(conf-lldp)# multiplier 2
```

To set the number of consecutive misses for a specific LLDP profile:

```
device(conf-lldp)# profile test1
device(config-profile-test1)# multiplier 5
device(config-profile-test1)#
```

## Related Commands

[hello-interval \(ELD\)](#)

# multiplier (UDLD)

Sets timeout multiplier for missed UDLD PDUs.

## Syntax

`multiplier value`

`no multiplier`

## Command Default

Multiplier default value is 5.

## Parameters

*value*

Specifies a multiplier value to use. Valid values range from 3 through 10.

## Modes

Protocol UDLD configuration mode

## Usage Guidelines

When the device at one end is an Extreme Networks IP product, the timeout interval is the product of the "hello" time interval at the other end and the "multiplier" value.

When the UDLD protocol times out waiting for UDLD PDUs, it will block the port.

Enter **no multiplier** to return to the default setting.

## Examples

To set the multiplier to 8:

```
device# configure terminal
device(config)# protocol udld
device(config-udld)# multiplier 8
```

## History

Release version	Command history
16r.1.00	This command was introduced.

## Related Commands

[hello \(UDLD\)](#)

# name (VLAN interfaces)

Assigns a descriptive name to a VLAN. Although this name cannot be used in place of the *vlan\_ID*, it is displayed in response to the **show vlan brief** command.

## Syntax

**name** *vlan\_name*

**no name**

## Parameters

*vlan\_name*

Specifies the characters of the name. The string must be between 1 and 32 characters.

## Modes

VLAN interface sub-type

## Usage Guidelines

If no name is assigned to a VLAN, a default name is automatically assigned, composed of "VLAN" and the *vlan\_ID*. For example, if the *vlan\_ID* is 1000, the default name is VLAN1000.

To revert from an assigned name to the default name, enter **no name**.

## Examples

Assign the name "marketing" to VLAN 1000:

```
device# configure
device(config)# interface vlan 1000
device(config-Vlan-1000)# name marketing
device(config-Vlan-1000)
```

## History

Release version	Command history
5.0.1	This command was introduced.

## Related Commands

[show vlan brief](#)

## nas auto-qos

Enables the Quality of Service (QoS) functionality for the Auto-NAS (automatic network attached storage) feature.

### Syntax

`nas auto-qos`

`no nas auto-qos`

### Modes

Global configuration mode

### Usage Guidelines

This command is supported only on Extreme VDX 8770-4, VDX 8770-8, VDX 6740, and VDX 6740T devices.

Use the **no** form of this command to disable the Auto-NAS feature.

# nas server-ip

Identifies the port that is to receive Auto NAS (automatic network attached storage) traffic.

## Syntax

```
nas server-ip address/prefix [ vlan vlan_ID | vrf vrf_name ]  
no nas server-ip address
```

## Parameters

*address/prefix*  
IP address/prefix to receive NAS traffic.

**vlan** *vlan\_ID*  
VLAN ID.

**vrf** *vrf\_name*  
VRF name.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on Extreme VDX 8770-4, VDX 8770-8, VDX 6740, and VDX 6740T devices.

Use the **no** form of this command to remove a nas server-ip prefix.

## Examples

To identify an IP address/prefix of 2.2.2.2/32 to receive NAS traffic over VLAN 10:

```
switch# configure  
switch(config)# nas server-ip 2.2.2.2/32 vlan 10
```

# nbr-timeout

Configures the neighbor timeout interval after which a neighbor is considered to be absent.

## Syntax

```
nbr-timeout num  
no nbr-timeout
```

## Command Default

The default is 105 seconds.

## Parameters

*num*  
Interval value in seconds. Valid values range from 35 through 12600 seconds.

## Modes

Router PIM configuration mode

## Usage Guidelines

Neighbor timeout is the interval after which a PIM device will consider a neighbor to be absent. Absence of PIM hello messages from a neighboring device indicates that a neighbor is not present. The interval can be set between 3 and 65535 seconds, and it should not be less than 3.5 times the hello timer value.

Enter **no nbr-timeout** to disable this feature.

## Examples

Setting the timeout to 600 seconds.

```
device(config)# router pim  
device(config-pim-router)# nbr-timeout 600
```

## Related Commands

[router pim](#)



# neighbor (OSPF)

Manually configures a neighbor.

## Syntax

```
neighbor A.B.C.D
```

```
no neighbor
```

## Command Default

Neighbors are not configured.

## Parameters

*A.B.C.D*

IPv4 address of the neighbor.

## Modes

OSPF router VRF configuration mode

## Usage Guidelines

This command is typically used in non-broadcast networks.

OSPF Hellos must use a unicast address, not broadcast or multicast packets.

Enter **no neighbor***A.B.C.D* to remove the specified neighbor.

## Examples

The following example configures a neighbor whose IPv4 address is 1.1.1.1.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# neighbor 1.1.1.1
```

# neighbor accept-lldp-neighbors

Enables BGP automatic neighbor discovery.

## Syntax

```
neighbor peer-group-name accept-lldp-neighbors
```

```
no neighbor peer-group-name accept-lldp-neighbors
```

## Command Default

BGP automatic neighbor discovery is disabled by default.

## Parameters

*peer-group-name*

Specifies the peer group name. The properties of this peer group are applied to the new BGP automatically discovered neighbors using LLDP.

## Modes

BGP configuration mode

## Usage Guidelines

BGP automatic neighbor discovery is only enabled when an MD5 password is configured for the peer group. Refer to the **neighbor password** command for more information. The **no** form of the command disables BGP automatic neighbor discovery.

## Examples

The following example enables BGP automatic neighbor discovery for a peer group called "mypg1". An MD5 password is also configured for the peer-group.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# local-as 100
device(config-bgp-router)# neighbor mypg1 peer-group
device(config-bgp-router)# neighbor mypg1 accept-lldp-neighbors
device(config-bgp-router)# neighbor mypg1 password extreme
```

## History

Release version	Command history
7.2.0	This command was introduced.

# neighbor activate

Enables the exchange of information with BGP neighbors and peer groups.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } activate
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } activate
```

## Command Default

Enabled under the IPv4 address family.

Disabled under the IPv6 address family.

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor.

*ipv6-address*

Specifies the IPv6 address of the neighbor.

*peer-group-name*

Specifies a peer group.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

To negotiate only the BGP L2VPN EVPN address family on an IPv6 neighbor, you must activate the neighbor only under the BGP L2VPN EVPN address family.

The **no** form of the command to disables the exchange of an address with a BGP neighbor or peer group. To negotiate only the BGP L2VPN EVPN address family on an IPv4 neighbor, you must explicitly deactivate the IPv4 unicast address family using the **no** form of the command.

## Examples

The following example establishes a BGP session with a neighbor with the IPv6 address 2001:2018:8192::125.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 activate
```

The following example establishes a BGP session with a neighbor with the IPv6 address 2001:2018:8192::125 for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 activate
```

The following example establishes a BGP session with a neighbor with the IP address 10.1.1.1 in L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.1.1.1 activate
```

The following example deactivates a neighbor under the IPv4 unicast address family and negotiates only the BGP L2VPN EVPN address family on an IPv4 neighbor.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device (config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# no neighbor 10.1.1.2 activate
device(config-bgp-ipv4u)# exit
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.1.1.2 activate
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.

## Related Commands

[address-family l2vpn evpn](#), [address-family unicast \(BGP\)](#), [router bgp](#)

# neighbor additional-paths advertise

Applies filters for the advertisement of additional paths for BGP neighbors.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } additional-paths advertise { all | best number }
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } additional-paths advertise { all | best number }
```

## Command Default

Disabled.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor .

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

**all**

Advertises all BGP additional paths with a unique next hop.

**best**

Advertises the additional paths that the device selects as best paths.

*value*

Specifies the number of best paths advertised. Valid values range from 1 through 5.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Only ECMP paths are considered for selection. The **best value** parameter currently advertises the best path plus the first 5 available ECMP paths. The **no** form of the command disables the configured filter.

## Examples

The following example configures BGP4 to advertise all BGP additional paths.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 additional-paths advertise all
```

The following example configures BGP4 to advertise all BGP additional paths for VRF "green".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf green
device(config-bgp-ipv4u-vrf)# neighbor 10.11.12.13 additional-paths advertise all
```

The following example configures BGP4+ advertise 4 best BGP additional paths.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 additional-paths advertise best 4
```

The following example configures BGP4+ advertise 3 best BGP additional paths for VRF "green".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf green
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 additional-paths advertise best 3
```

## History

Release version	Command history
7.0.0	This command was introduced.

# neighbor advertisement-interval

Enables changes to the interval over which a specified neighbor or peer group holds route updates before forwarding them.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **advertisement-interval** *seconds*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **advertisement-interval**

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

*seconds*

Range is from 0 through 3600. The default is 0.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the default interval.

## Examples

The following example changes the BGP4 advertisement interval from the default to 60 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 advertisement-interval 60
```

The following example changes the BGP4+ advertisement interval from the default for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 advertisement-interval 60
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.



# neighbor allowas-in

Disables the AS\_PATH check function for routes learned from a specified neighbor so that BGP does not reject routes that contain the recipient BGP speaker's AS number.

## Syntax

```
neighbor {ip-address | ipv6-address | peer-group-name } allowas-in number
no neighbor allowas-in {ip-address | ipv6-address | peer-group-name } allowas-in
```

## Command Default

The AS\_PATH check function is enabled and any route whose path contains the speaker's AS number is rejected as a loop.

## Parameters

*ip-address*

Specifies the IP address of the neighbor.

*ipv6-address*

Specifies the IPv6 address of the neighbor.

*peer-group-name*

Specifies a peer group.

*number*

Specifies the number of times that the AS path of a received route may contain the recipient BGP speaker's AS number and still be accepted. Valid values are 1 through 10.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

The **no** form of the command re-enables the AS\_PATH check function.

If the AS\_PATH check function is disabled after a BGP session has been established, the neighbor session must be cleared for this change to take effect.

## Examples

The following example specifies that the AS path of a received route may contain the recipient BGP4+ speaker's AS number three times and still be accepted.

```
device#configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 allowas-in 3
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

The following example specifies for VRF instance "red" that the BGP4+ AS path of a received route may contain the recipient BGP speaker's AS number three times and still be accepted.

```
device#configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::124 allowas-in 3
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

The following example specifies that the AS path of a received route may contain the recipient BGP speaker's AS number three times and still be accepted in L2VPN EVPN configuration mode.

```
device#configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.1.1.1 allowas-in 3
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.

## Related Commands

[address-family l2vpn evpn](#), [address-family unicast \(BGP\)](#), [router bgp](#)

# neighbor alternate-as

Adds a range of alternate autonomous system numbers (ASNs) for BGP dynamic neighbors.

## Syntax

```
neighbor peer-group-name alternate-as { add | remove } as-range
```

```
no neighbor peer-group-name alternate-as { add | remove } as-range
```

## Command Default

Disabled.

## Parameters

### add

Adds an AS to the alternate AS range.

### remove

Removes an AS from the alternate AS range.

### *as-range*

Specifies an alternate AS value. Enter an integer from 1 through 4294967295.

## Modes

BGP configuration mode

## Usage Guidelines

This command supports only IPv4 BGP.

The **no** form of the command removes configured alternate AS values.

## Examples

The following example sets an alternate AS of 100 for listen range neighbors in a peer group called "mypeergroup".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor mypeergroup remote-as 400
device(config-bgp-router)# neighbor mypeergroup alternate-as add 100
```

The following example sets an alternate AS range of 200 through 300 for listen range neighbors in a peer group called "mypeergroup".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor mypeergroup remote-as 400
device(config-bgp-router)# neighbor mypeergroup alternate-as add 200-300
```

The following example removes the configured alternate AS number 100 for listen range neighbors in a peer group called "mypeergroup".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor mypeergroup alternate-as remove 100
```

## History

Release version	Command history
7.1.0	This command was introduced.

# neighbor as-override

Replaces the autonomous system number (ASN) of the originating device with the ASN of the sending BGP device.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } as-override
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } as-override
```

## Command Default

This feature is disabled.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the `neighbor peer-group-name` command.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to disable this feature.

BGP loop prevention verifies the ASN in the AS path. If the receiving router sees its own ASN in the AS path of the received BGP packet, the packet is dropped. The receiving router assumes that the packet originated from its own AS and has reached the place of origination. This can be a significant problem if the same ASN is used among various sites, preventing sites with identical ASNs from being linked by another ASN. In this case, routing updates are dropped when another site receives them.

## Examples

This example replaces the ASN globally.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 as-override
```

This example replaces the BGP4+ ASN for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 as-override
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor bfd

Enables BFD sessions for specified BGP neighbors or peer groups.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **bfd** [ **holdover-interval** *time* | **interval** *transmit-time* **min-rx** *receive-time* **multiplier** *number* ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **bfd** [ **holdover-interval** *time* | **interval** *transmit-time* **min-rx** *receive-time* **multiplier** *number* ]

## Command Default

BFD sessions are not enabled on specific BGP neighbors or peer groups.

## Parameters

*ip-address*

Specifies the IP address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Specifies a peer group.

**holdover-interval** *time*

Specifies the holdover interval, in seconds, for which BFD session down notifications are delayed before notification that a BFD session is down. Valid values range from 1 through 30. The default is 0.

**interval** *transmit-time*

Specifies the interval, in milliseconds, a device waits to send a control packet to BFD peers. On Extreme VDX 8770-4, VDX 8770-8, and later devices, valid values range from 50 through 30000, and the default is 500. On Extreme VDX 8770 platforms, valid values range from 50 to 30000, and the default is 200.

**min-rx** *receive-time*

Specifies the interval, in milliseconds, a device waits to receive a control packet from BFD peers. On Extreme VDX 6740, VDX 6740T, and VDX 6940 platforms, valid values range from 50 through 30000, and the default is 500. On Extreme VDX 8770 platforms, valid values range from 50 to 30000, and the default is 200.

**multiplier**

Specifies the number of consecutive BFD control packets that must be missed from a BFD peer before BFD determines that the connection to that peer is not operational. Valid values range from 3 through 50. The default is 3.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Before using the **holdover-interval**, **interval**, **min-rx**, and **multiplier** parameters, you must first enable BFD using the **neighbor** `{ip-address | peer-group-name} bfd` command.

For single hop BFD sessions, BFD considers the interval values that are configured on the interface, but not the non-default values that are configured with this global command.

The **no** form of the command removes the BFD for BGP configuration for BGP neighbors or peer groups.

## Examples

The following example configures BFD for a specified peer group and sets the BFD holdover interval to 18.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# neighbor pgl bfd
device(config-bgp-router)# neighbor pgl bfd holdover-interval 18
```

The following example configures BFD for a specified peer group and sets the BFD holdover interval 12 for VRF instance "green".

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf greene
device(config-bgp-ipv4u-vrf)# neighbor pgl bfd
device(config-bgp-ipv4u-vrf)# neighbor pgl bfd holdover-interval 12
```

The following example configures BFD for a BGP neighbor with the IP address 10.10.1.1 and sets the BFD session timer values.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# neighbor 10.10.1.1 bfd
device(config-bgp-router)# neighbor 10.10.1.1 bfd interval 120 min-rx 150 multiplier 8
```

## History

Release version	Command history
6.0.1	This command was introduced.



# neighbor capability additional-paths

Enables the advertisement of additional paths for BGP neighbors.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **capability additional-paths** [ **receive** | **send** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **capability additional-paths** [ **receive** | **send** ]

## Command Default

Additional paths are not advertised.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor .

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

**receive**

Enables BGP to receive additional paths from BGP neighbors.

**send**

Enables BGP to send additional paths to BGP neighbors.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command disables the advertisement of additional-paths for BGP neighbors. If neither the **send** or **receive** parameter is used, the send and receive capabilities are applied.

## Examples

The following example enables BGP4 to send additional paths to all BGP neighbors.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 capability additional-paths send
```

The following example enables BGP4 to send additional paths to all BGP neighbors for VRF "green".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf green
device(config-bgp-ipv4u-vrf)# neighbor 10.11.12.13 capability additional-paths send
```

The following example enables BGP4+ receive additional paths from all BGP neighbors.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 capability additional-paths receive
```

The following example enables BGP4+ to send additional paths to all BGP neighbors for VRF "green".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf green
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 capability additional-paths send
```

## History

Release version	Command history
7.0.0	This command was introduced.

# neighbor capability as4

Enables or disables support for 4-byte autonomous system numbers (ASNs) at the neighbor or peer-group level.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } capability as4 [ disable | enable ]
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } capability as4 [ disable | enable ]
```

## Command Default

4-byte ASNs are disabled by default.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor .

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

**disable**

Disables 4-byte numbering.

**enable**

Enables 4-byte numbering.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **disable** keyword or the **no** form of this command to remove all neighbor capability for 4-byte ASNs.

4-byte ASNs are first considered at the neighbor, then at the peer group, and finally at the global level.

## Examples

This example enables 4-byte ASNs globally.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 capability as4 enable
```

This example enables BGP4+ 4-byte ASNs on VRF instance "red":

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 capability as4 enable
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor capability orf prefixlist

Advertises outbound route filter (ORF) capabilities to peer routers.

## Syntax

```
neighbor { ip_address | ipv6_address | peer-group-name } capability orf prefixlist [ receive | send ]
```

```
no neighbor { ip_address | ipv6_address | peer-group-name } capability orf prefixlist [ receive | send ]
```

## Command Default

ORF capabilities are not advertised to a peer device.

## Parameters

*ip\_address*

Specifies the IPv4 address of the neighbor.

*ipv6\_address*

Specifies the IPv6 address of the neighbor.

*peer-group-name*

Specifies a peer group.

**receive**

Enables the ORF prefix list capability in receive mode.

**send**

Enables the ORF prefix list capability in send mode.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to disable ORF capabilities.

## Examples

This example advertises the ORF send capability to a neighbor with the IP address 10.11.12.13.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 capability orf prefixlist send
```

This example advertises the ORF receive capability to a neighbor with the IPv6 address 2001:2018:8192::125 for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 capability orf prefixlist receive
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor default-originate

Configures the device to send the default route 0.0.0.0 to a neighbor.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } default-originate [ route-map map-name ]
no neighbor { ip-address | ipv6-address | peer-group-name } default-originate [ route-map map-name ]
```

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

**route-map**

Optionally injects the default route conditionally, depending on the match conditions in the route map.

*map-name*

Name of the route map.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

## Examples

This example sends the default route to a BGP4 neighbor for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 default-originate route-map myroutemap
```

This example sends the default route to a BGP4+ neighbor for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 default-originate route-map myroutemap
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

## Related Commands

[route-map](#)



# neighbor description

Specifies a name for a neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **description** *string*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **description**

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

**description** *string*

Specifies the name of the neighbor, an alphanumeric string up to 220 characters long.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes the name.

## Examples

The following example specifies a BGP4 neighbor name.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 description mygoodneighbor
```

The following example specifies a BGP4+ neighbor name for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 default-originate route-map myroutemap
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor ebgp-btsh

Enables BGP time to live (TTL) security hack protection (BTSH) for eBGP.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **ebgp-btsh**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **ebgp-btsh**

## Command Default

Disabled.

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor.

*ipv6-address*

Specifies the IPv6 address of the neighbor.

*peer-group-name*

Specifies a peer group.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

To maximize the effectiveness of this feature, the **neighbor ebgp-btsh** command should be executed on each participating device. The **neighbor ebgp-btsh** command is supported for both directly connected peering sessions and multihop eBGP peering sessions. For directly connected neighbors, when the **neighbor ebgp-btsh** command is used, the device expects BGP control packets received from the neighbor to have a TTL value of either 254 or 255. For multihop peers, when the **neighbor ebgp-btsh** command is used, the device expects the TTL for BGP control packets received from the neighbor to be greater than or equal to 255 minus the configured number of hops to the neighbor.

The **no** form of the command disables BTSH for eBGP.

## Examples

The following example enables GTSM between a device and a neighbor with the IP address 10.10.10.1.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.1.1.1 ebgp-btsh
```

neighbor ebgp-btsh

The following example enables GTSM between a device and a neighbor with the IPv6 address 2001:2018:8192::125.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# neighbor 2001:2018:8192::125 ebgp-btsh
```

## History

Release version	Command history
7.0.0	This command was introduced.

# neighbor ebgp-multihop

Allows eBGP neighbors that are not on directly connected networks and sets an optional maximum hop count.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } ebgp-multihop [ max-hop-count ]
no neighbor { ip-address | ipv6-address | peer-group-name } ebgp-multihop
```

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

*max-hop-count*

Maximum hop count. Range is from 1 through 255.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When eBGP is used in conjunction with the IP unnumbered interfaces feature, it is important to ensure that sufficient Time To Live (TTL) values are available for hello messages to establish separate neighbor adjacencies on leaf switches in an IP Fabric. To do so, use the **neighbor ebgp-multihop** command and set the number of maximum hops to **2**.

## Examples

The following example enables eBGP multihop and sets the maximum hop count to 20.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 ebgp-multihop 20
```

The following example enables BGP4+ eBGP multihop for VRF instance "red" and sets the maximum hop count to 40.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 ebgp-multihop 40
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.1	This command was modified to include Usage Guidelines when used in conjunction with the <b>neighbor ebgp-multipath</b> command.

# neighbor enable-peer-as-check

Enables the outbound AS\_PATH check function so that a BGP sender speaker does not send routes with an AS path that contains the ASN of the receiving speaker.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } enable-peer-as-check
no neighbor { ip-address | ipv6-address | peer-group-name } enable-peer-as-check
```

## Command Default

Disabled.

## Parameters

*ip-address*  
Specifies the IPv4 address of the neighbor.

*ipv6-address*  
Specifies the IPv6 address of the neighbor.

*peer-group-name*  
Specifies a peer group.

## Modes

BGP address-family IPv4 unicast configuration mode  
 BGP address-family IPv6 unicast configuration mode  
 BGP address-family IPv4 unicast VRF configuration mode  
 BGP address-family IPv6 unicast VRF configuration mode  
 BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

When the **neighbor enable-peer-as-check** command is used for a BGP address family, a neighbor reset is required.

The **no** form of the command disables the AS-path check function.

## Examples

The following example enables the outbound AS\_PATH check function for the BGP IPv4 unicast address family.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.1.1.1 enable-peer-as-check
```

The following example enables the outbound AS\_PATH check function for the BGP IPv6 unicast address family.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 enable-peer-as-check
```

The following example enables the outbound AS\_PATH check function for the L2VPN EVPN unicast address family.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.1.1.1 enable-peer-as-check
```

## History

Release version	Command history
7.0.1	This command was introduced.



# neighbor enforce-first-as

Ensures that a device requires the first ASN listed in the AS\_SEQUENCE field of an AS path-update message from EBGP neighbors to be the ASN of the neighbor that sent the update.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } enforce-first-as [ disable | enable ]
no neighbor { ip-address | ipv6-address | peer-group-name } enforce-first-as [ disable | enable ]
```

## Command Default

Disabled by default.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

**disable**

Disables this feature.

**enable**

Enables this feature.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to disable this requirement globally for the device.

## Examples

This example enables the enforce-first-as feature for a specified neighbor.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 enforce-first-as enable
```

This example enables the enforce-first-as feature for a BGP4+ specified neighbor for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 enforce-first-as enable
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor filter-list

Specifies a filter list to be applied to updates from or to the specified neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **filter-list** *ip-prefix-list-name* { **in** | **out** }

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **filter-list** *ip-prefix-list-name* { **in** | **out** }

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

*ip-prefix-list-name*

Name of the filter list.

**in**

Specifies that the list is applied on updates received from the neighbor.

**out**

Specifies that the list is applied on updates sent to the neighbor.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

## Examples

This BGP4 example specifies that filter list "myfilterlist" be applied to updates to a neighbor with the IP address 10.11.12.13 for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 filter-list myfilterlist out
```

This BGP4+ example specifies that filter list "2" be applied to updates from a neighbor with the IPv6 address 2001:2018:8192::125 for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 filter-list 2 in
```

This BGP4+ example specifies that filter list "2" be applied to updates from a neighbor with the IPv6 address 2001:2018:8192::125 for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 filter-list 2 in
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor graceful-shutdown

Gracefully shuts down a BGP neighbor or peer group.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **graceful-shutdown** *seconds* [ **community** *value* [ **local-preference** *value* ] ] | **local-preference** *value* [ **community** *value* ] ] | **route-map** *route-map-name* ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **graceful-shutdown** *seconds* [ **community** *value* [ **local-preference** *value* ] ] | **local-preference** *value* [ **community** *value* ] ] | **route-map** *route-map-name* ]

## Command Default

Default graceful shutdown parameters are applied.

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor

*ipv6-address*

Specifies the IPv6 address of the neighbor

*peer-group-name*

Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

*seconds*

Specifies the number of seconds in which the BGP graceful shutdown will occur. Valid values range from 30 through 600 seconds.

**community** *value*

Sets the community attribute for graceful shutdown. Valid values range from 1 through 4294967295.

**local-preference** *value*

Sets the local preference attribute for graceful shutdown. Valid values range from 0 through 4294967295.

**route-map** *route-map-name*

Specifies the route map for graceful shutdown attributes.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command de-activates graceful shutdown.

## Examples

The following example gracefully shuts down the neighbor 10.11.22.23 and sets the graceful shutdown timer to 580 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.22.23 graceful-shutdown 580
```

The following example gracefully shuts down the peer group "grp-1" and sets the graceful shutdown timer to 620 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor grp-1 graceful-shutdown 620
```

The following example gracefully shuts down the neighbor 10.11.22.23 and sets the graceful shutdown timer to 600 seconds.

The route map "myroutemap" is specified for graceful shutdown attributes.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.22.23 graceful-shutdown 600 route-map myroutemap
```

The following example gracefully shuts down the neighbor 10.11.22.23 and sets the graceful shutdown timer to 600 seconds.

The community attribute is set to 20.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.22.23 graceful-shutdown 600 community 20
```

## History

Release version	Command history
7.2.0	This command was introduced.

# neighbor local-as

Causes the device to prepend the local autonomous system number (ASN) automatically to routes received from an eBGP peer.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } local-as num [ no-prepend ]
no neighbor { ip-address | ipv6-address | peer-group-name } local-as num [ no-prepend ]
```

## Command Default

This feature is disabled.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

*num*

Local ASN. Range is from 1 through 4294967295.

**no-prepend**

Causes the device to stop prepending the selected ASN.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the local ASN.

## Examples

This example ensures that a device prepends the local ASN.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 local-as 100
```

This example stops the device from prepending the selected ASN.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 local-as 100 no-prepend
```

This example stops the device from prepending the selected ASN for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 local-as 100 no-prepend
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.



# neighbor maxas-limit in

Causes the device to discard routes received in UPDATE messages if those routes exceed a maximum AS path length.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } maxas-limit in { num | disable }
no neighbor { ip-address | ipv6-address | peer-group-name } maxas-limit in
```

## Command Default

This command is disabled by default.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

*num*

Maximum length of the AS path. Range is from 0 through 300. The default is 300.

**disable**

Prevents a neighbor from inheriting the configuration from the peer group or global configuration and instead uses the default system value.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to remove this configuration.

## Examples

This example changes the length of the maximum allowed AS path length from the default.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 maxas-limit in 200
```

This example prevents a neighbor from inheriting the configuration from the peer group or global configuration and instead use the default system value.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 2001:2018:8192::125 maxas-limit in disable
```

This BGP4+ example changes the length of the maximum allowed AS path length from the default for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 maxas-limit in 200
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor maximum-prefix

Specifies the maximum number of IP network prefixes (routes) that can be learned from a specified neighbor or peer group.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **maximum-prefix** *num* [ *threshold* ] [ **teardown** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **maximum-prefix** *num* [ *threshold* ] [ **teardown** ]

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor.

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

*num*

Maximum number of IP prefixes that can be learned. Range is from 0 through 4294967295. Default is 0 (unlimited).

*threshold*

Specifies the percentage of the value specified by *num* that causes a syslog message to be generated. Range is from 1 through 100. Default is 100.

**teardown**

Tears down the neighbor session if the maximum number of IP prefixes is exceeded.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

The *threshold* and **teardown** options are not available in BGP address-family L2VPN EVPN configuration mode.

The **no** form of the command restores the defaults.

## Examples

The following example sets the maximum number of prefixes that will be accepted from the neighbor with the IP address 10.11.12.13 to 100000, and sets the threshold value to 80%.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 maximum-prefix 100000 threshold 80
```

The following example, for VRF instance "red," sets the maximum number of prefixes that will be accepted from the neighbor with the IPv6 address 2001:2018:8192::125 to 100000, and sets the threshold value to 90%.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 maximum-prefix 100000 threshold 90
```

The following example sets the maximum number of prefixes that will be accepted from the neighbor with the IP address 10.1.2.3 to 100000 in L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.1.2.3 maximum-prefix 100000
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.

## Related Commands

[address-family l2vpn evpn](#), [address-family unicast \(BGP\)](#), [router bgp](#)

# neighbor next-hop-self

Causes the device to list itself as the next hop in updates that are sent to the specified neighbor.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } next-hop-self [ always ]
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } next-hop-self [ always ]
```

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor.

*ipv6-address*

Specifies the IPv6 address of the neighbor.

*peer-group-name*

Specifies the peer group name configured by the **neighbor peer-group-name** command.

**always**

Enables this feature for route reflector (RR) routes.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command disables this feature.

## Examples

The following example causes all updates destined for the neighbor with the IP address 10.11.12.13 to advertise this device as the next hop.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 next-hop-self
```

The following example, for the VRF instance "red," causes all updates destined for the neighbor with the IPv6 address 2001:2018:8192::125 to advertise this device as the next hop.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 next-hop-self
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor next-hop-unchanged

Enables BGP to send updates to eBGP peers with the next-hop attribute unchanged.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **next-hop-unchanged**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **next-hop-unchanged**

## Command Default

Enabled.

## Parameters

*ip-address*

Specifies the IP address of the neighbor.

*ipv6-address*

Specifies the IPv6 address of the neighbor.

*peer-group-name*

Specifies a peer group.

## Modes

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

By default, BGP speakers change the next hop while sending the updates to eBGP neighbors. Use the **neighbor next-hop-unchanged** command to override this behavior. When this command is used, the next hop attribute remains unchanged while updates are sent to eBGP peers, and the BGP speaker is forced to retain the next hop address in the BGP updates received from neighbors.

The **no** form of the command disables the sending of updates to eBGP peers with the next-hop attribute unchanged.

## Examples

The following example disables the sending of updates to eBGP peers with the next-hop attribute unchanged.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# no neighbor 10.11.12.13 next-hop-unchanged
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[address-family l2vpn evpn](#), [router bgp](#)



# neighbor password

Specifies an MD5 password for securing sessions between the device and a neighbor.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } password string
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } password
```

## Command Default

No password is set.

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor.

*ipv6-address*

Specifies the IPv6 address of the neighbor.

*peer-group-name*

Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

*string*

Password of up to 63 characters in length that can contain any alphanumeric character.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes a configured MD5 password.

## Examples

The following example specifies a password for securing sessions with a specified neighbor.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 password s0M3P@55W0Rd
```

The following BGP4+ example, for VRF instance "red," specifies a password for securing sessions with a specified neighbor.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv4u-vrf)# neighbor 2001:2018:8192::125 password s0M3P@55W0Rd
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor peer-group

Configures a BGP neighbor to be a member of a peer group.

## Syntax

```
neighbor { ip-address | ipv6-address } peer-group string
no neighbor { ip-address | ipv6-address } peer-group string
```

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor.

*ipv6-address*

Specifies the IPv6 address of the neighbor.

**peer-group** *string*

Specifies the name of a BGP peer group. The name can be up to 63 characters in length and can be composed of any alphanumeric character.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes a neighbor from the peer group.

## Examples

The following example assigns a specified neighbor to a peer group called "mypeergroup1".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 peer-group mypeergroup1
```

The following BGP4+ example, for VRF instance "red," assigns a specified neighbor to a peer group called "mypeergroup1".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 peer-group mypeergroup1
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor prefix-list

Filters the outgoing and incoming route updates to or from a particular BGP neighbor according to IP address and mask length.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } prefix-list string { in | out }
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } prefix-list string { in | out }
```

## Command Default

This feature is disabled.

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

*string*

Name of the prefix list. Range is from 1 through 63 ASCII characters.

**in**

Applies the filter in incoming routes.

**out**

Applies the filter in outgoing routes.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

## Examples

This example applies the prefix list "myprefixlist" to incoming advertisements to neighbor 10.11.12.13 for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 prefix-list myprefixlist in
```

This example applies the prefix list "myprefixlist" to outgoing advertisements to neighbor 2001:2018:8192::125 for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 prefix-list myprefixlist out
```

This example applies the prefix list "myprefixlist" to outgoing advertisements to neighbor 2001:2018:8192::125 for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 prefix-list myprefixlist out
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor remote-as

Specifies the autonomous system (AS) in which a remote neighbor resides.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } remote-as num
no neighbor { ip-address | ipv6-address | peer-group-name } remote-as
```

## Command Default

No AS is specified.

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

*num*

Remote AS number (ASN). Range is from 1 through 4294967295.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes the neighbor from the AS.

## Examples

The following example specifies AS 100 for a neighbor.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 remote-as 100
```

neighbor remote-as

The following BGP4+ example, for VRF instance "red," specifies AS 100 for a neighbor.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 remote-as 100
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.



# neighbor remove-private-as

Configures a device to remove private autonomous system numbers (ASNs) from UPDATE messages that the device sends to a neighbor.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } remove-private-as
no neighbor { ip-address | ipv6-address | peer-group-name } remove-private-as
```

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The device will remove ASNs 64512 through 65535 (the well-known BGP4 private ASNs) from the AS-path attribute in UPDATE messages that the device sends to a neighbor.

The **no** form of the command restores the default so that private ASNs are not removed from UPDATE messages sent to a neighbor by a device.

## Examples

The following example removes private ASNs globally.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 remove-private-as
```

The following example removes private ASNs for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 remove-private-as
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor route-map

Filters the outgoing and incoming route updates to or from a particular BGP neighbor according to a set of attributes defined in a route map.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } route-map { in string | out string }
no neighbor { ip-address | ipv6-address | peer-group-name } route-map { in string | out string }
```

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

**in**

Applies the filter on incoming routes.

*string*

Name of the route map.

**out**

Applies the filter on outgoing routes.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

The **no** form of the command restores the defaults.

## Examples

The following example applies a route map named "myroutemap" to an outgoing route from 10.11.12.13.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 route-map out myroutemap
```

The following example applies a route map named "myroutemap" to an incoming route from 2001:2018:8192::125.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 route-map in myroutemap
```

The following example applies a route map named "myroutemap" to an outgoing route from 10.11.12.13 in L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.11.12.13 route-map out myroutemap
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.

## Related Commands

[address-family l2vpn evpn](#), [address-family unicast \(BGP\)](#), [route-map](#), [router bgp](#)

# neighbor route-reflector-client

Configures a neighbor to be a route-reflector client.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **route-reflector-client**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **route-reflector-client**

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

Use this command on a host device to configure a neighbor to be a route-reflector client. Once configured, the host device from which the configuration is made acts as a route-reflector server.

The **no** form of the command restores the default.

## Examples

The following example configures a neighbor to be a route-reflector client.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 route-reflector-client
```

The following example configures a neighbor to be a route-reflector client for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv4u-vrf)# neighbor 2001:2018:8192::125 route-reflector-client
```

The following example configures a neighbor to be a route-reflector client in L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.1.1.1 route-reflector-client
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.

## Related Commands

[address-family l2vpn evpn](#), [address-family unicast \(BGP\)](#), [router bgp](#)

# neighbor send-community

Enables sending the community attribute in updates to the specified BGP neighbor.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } send-community [ both | extended | standard ]
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } send-community [ both | extended | standard ]
```

## Command Default

The device does not send community attributes.

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

**both**

Sends both standard and extended attributes.

**extended**

Sends extended attributes.

**standard**

Sends standard attributes.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

The **no** form of the command restores the defaults.

If the **send-community** attribute is enabled after a BGP session has been established, the neighbor session must be cleared for this change to take effect.

## Examples

The following example sends standard community attributes to a neighbor.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 send-community standard
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

The following example sends extended community attributes to a neighbor for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 send-community extended
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

The following example sends standard and extended community attributes to a neighbor in L2VPN EVPN configuration mode.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# neighbor 10.1.1.1 send-community both
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	Support was added for the BGP address-family L2VPN EVPN configuration mode.



# neighbor shutdown

Causes a device to shut down the session administratively with its BGP neighbor.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } shutdown [ generate-rib-out ]
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } shutdown [ generate-rib-out ]
```

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor** *peer-group-name* command.

**generate-rib-out**

When a peer is put into the shutdown state, Routing Information Base (RIB) outbound routes are not produced for that peer. Use this option to produce those routes.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Shutting down a session lets you configure the neighbor and save the configuration without the need to establish a session with that neighbor.

## Examples

The following example causes a device to shut down globally the session administratively with its neighbor and generate RIB outbound routes.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 shutdown generate-rib-out
```

The following example causes a device to shut down the session administratively with its neighbor and generate RIB outbound routes for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 shutdown generate-rib-out
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

## Related Commands

[auto-shutdown-new-neighbors](#)

# neighbor soft-reconfiguration inbound

Stores all the route updates received from a BGP neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **soft-reconfiguration inbound**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **soft-reconfiguration inbound**

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor

*ipv6-address*

Specifies the IPv6 address of the neighbor

*peer-group-name*

Specifies the peer group name.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Soft reconfiguration stores all the route updates received from a neighbor. If you request a soft reset of inbound routes, the software compares the policies against the stored route updates, instead of requesting the neighbor's BGP4 or BGP4+ route table or resetting the session with the neighbor.

## Examples

The following example globally stores route updates from a BGP4 neighbor.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 soft-configuration inbound
```

The following example stores route updates from a BGP4+ neighbor for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 soft-configuration inbound
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor static-network-edge

Overrides the default BGP4 behavior and advertises the network to a neighbor or peer group only when the corresponding route is installed as a forward route in the routing table.

## Syntax

```
neighbor { ip-address | peer-group-name } static-network-edge
no neighbor { ip-address | peer-group-name } static-network-edge
```

## Parameters

*ip-address*

Specifies the IPv4 address of the neighbor

*peer-group-name*

Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

## Usage Guidelines

A BGP static network is always advertised to neighbors or a peer group, and if the corresponding route is not present in the routing table, BGP installs the null0 route. This command overrides the default behavior. This command is not supported for BGP4+.

## Examples

The following example globally overrides the default BGP4 behavior.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 static-network-edge
```

The following example overrides the default BGP4 behavior for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# neighbor 10.11.12.13 static-network-edge
```

## History

Release version	Command history
6.0.1	This command was introduced.

# neighbor timers

Specifies how frequently a device sends KEEPALIVE messages to its BGP neighbors, as well as how long the device waits for KEEPALIVE or UPDATE messages before concluding that a neighbor is dead.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } timers keep-alive keepalive_interval hold-time holdtime_interval
no neighbor { ip-address | ipv6-address | peer-group-name } timers keep-alive keepalive_interval hold-time holdtime_interval
```

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

**keep-alive** *keepalive\_interval*

Frequency (in seconds) with which a device sends keepalive messages to a peer. Range is from 0 through 65535 seconds. The default is 60.

**hold-time** *holdtime\_interval*

Interval in seconds that a device waits to receive a keepalive message from a peer before declaring that peer dead. Range is from 0 through 65535 seconds. The default is 180.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the defaults.

## Examples

The following example sets the keepalive timer for a device to 120 seconds and the hold-timer to 360 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 timers keep-alive 120 hold-time 360
```

The following example sets the keepalive timer to 120 seconds and the hold-timer to 360 seconds for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 timers keep-alive 120 hold-time 360
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# neighbor unsuppress-map

Removes route suppression from BGP neighbor routes when those routes have been suppressed as a result of aggregation. All routes matching route-map rules are unsuppressed.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } unsuppress-map string
no neighbor { ip-address | ipv6-address | peer-group-name } unsuppress-map string
```

## Command Default

This feature is disabled.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

*string*

Name of the route map. Range is from 1 through 63 ASCII characters.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

## Examples

The following BGP4 example removes route suppression for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 unsuppress-map myroutemap
```



The following BGP4+ example removes route suppression for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 unsuppress-map myroutemap
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

## Related Commands

[route-map](#)

# neighbor update-source

Configures the BGP device to communicate with a neighbor through a specified interface.

## Syntax

```
neighbor { ip-address | ipv6-address | peer-group-name } update-source { ip-address | <N>gigabitethernet rbridge-id / slot / port | loopback num | port-channel number | ve-interface vlan_id }
```

```
no neighbor { ip-address | ipv6-address | peer-group-name } update-source { ip-address | <N>gigabitethernet rbridge-id / slot / port | loopback num | ve-interface vlan_id }
```

## Parameters

*ip-address*

IPv4 address of the neighbor

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Peer group name configured by the **neighbor peer-group-name** command.

*ip-address*

IP address of the update source.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**loopback** *num*

Specifies a loopback interface.

**port-channel** *number*

Specifies a port-channel interface. The range is from 1 through 6144.

**ve-interface** *vlan\_id*

Specifies a virtual Ethernet VLAN interface.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

## Examples

The following example configures the device to communicate with a neighbor through the specified IPv4 address and port.

```
device#configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 update-source tengigabitethernet 15/1/1
```

The following example configures the device to communicate, for VRF instance "red," with a neighbor through the specified IPv6 address and port.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 update-source tengigabitethernet 15/1/1
```

## History

Release version	Command history
5.0.0	This command was modified to add support for IPv6.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.
7.0.0	This command was modified to support port-channels.

# neighbor weight

Specifies a weight that the device will add to routes that are received from the specified BGP neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **weight** *num*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **weight**

## Command Default

The default for *num* is 0.

## Parameters

*ip-address*

IPv4 address of the neighbor.

*ipv6-address*

IPv6 address of the neighbor

*peer-group-name*

Name of the peer group.

*num*

Value from 1 through 65535.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

BGP prefers larger weights over smaller weights.

## Examples

This example changes the weight from the default.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# neighbor 10.11.12.13 weight 100
```

This example changes the weight from the default for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 weight 100
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# network

Configures the device to advertise a BGP network.

## Syntax

**network** *network/mask* [ **backdoor** | **route-map** *map-name* | **weight** *num* ]

**no network** *network/mask* [ **backdoor** | **route-map** *map-name* | **weight** *num* ]

## Command Default

No network is advertised.

## Parameters

*network/mask*

Network and mask in CIDR notation.

**backdoor**

Changes administrative distance of the route to this network from the EBGp administrative distance (the default is 20) to the local BGP4 weight (the default is 200), tagging the route as a backdoor route.

**route-map** *map-name*

Specifies a route map with which to set or change BGP4 attributes for the network to be advertised.

**weight***num*

Specifies a weight to be added to routes to this network. Range is 0 through 65535. The default is 0.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

## Examples

This example imports the IPv4 network 10.11.12.12/30 into the route map "myroutemap".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# network 10.11.12.13/30 route-map myroutemap
```

This example imports the IPv6 prefix 2001:db8::/32 into the BGP4+ database and sets a weight of 300.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# network 2001:db8::/32 weight 300
```

This example imports the IPv6 prefix 2001:db8::/32 into the BGP4+ database in VRF instance "red" and changes the administrative distance of the route to this network to the local BGP weight, tagging the route as a backdoor route.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# network 2001:db8::/32 backdoor
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

## Related Commands

[route-map](#)

# next-hop-enable-default

Configures the device to use the BGP default route as the next hop.

## Syntax

**next-hop-enable-default**

**no next-hop-enable-default**

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

This command is not available for a nondefault VRF instance. All nondefault VRFs configured under the IPv4 or IPv6 address-family unicast modes inherit their values from the default VRF.

## Examples

This BGP4 example configures the device to use the default route as the next hop for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# next-hop-enable-default
```

This BGP4+ example configures the device to use the default route as the next hop for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# next-hop-enable-default
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Usage guidelines were modified to indicate inheritance from the default VRF in support of Multi-VRF.



# next-hop-recursion

Enables BGP recursive next-hop lookups.

## Syntax

`next-hop-recursion`

`no next-hop-recursion`

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

If the BGP next hop is not the immediate next hop, a recursive route lookup in the IP routing information base (RIB) is needed. With recursion, a second routing lookup is required to resolve the exit path for destination traffic. Use this command to enable recursive next-hop lookups.

This command is not available for a nondefault VRF instance. All nondefault VRFs configured under the IPv4 or IPv6 address-family unicast modes inherit their values from the default VRF.

## Examples

This example enables recursive next-hop lookups for BGP4.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# next-hop-recursion
```

This example enables recursive next-hop lookups for the default VRF for BGP4+.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# next-hop-recursion
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Usage guidelines were modified to indicate inheritance from the default VRF in support of Multi-VRF.

# nonstop-routing (OSPF)

Enables nonstop routing (NSR) for OSPF.

## Syntax

**nonstop-routing**

**no nonstop-routing**

## Command Default

Enabled.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command disables non-stop routing.

## Examples

The following example re-enables NSR on a device.

```
device# configuration terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# nonstop-routing
```

## History

Release version	Command history
5.0.0	This command was introduced.

## Related Commands

[graceful-restart helper \(OSPFv3\)](#)

# nport interface fiberchannel (ag)

Under Access Gateway, accesses an N\_Port in order to map a specific VF\_Port to it.

## Syntax

```
nport interface fiberchannel port
```

## Command Default

If you do not map VF\_Ports to N\_Ports, VF\_Ports are assigned to available N\_Ports in a round-robin fashion, as ENodes log in.

## Parameters

*port*

Specifies the number of the N\_Port to which you want to map a VF\_Port. N\_Ports are identified by *rbridge-id/slot/N\_Port*, such as 3/0/4 for RBridge 3, slot 0, and N\_Port 4.

## Modes

Access Gateway configuration mode.

## Examples

The following example maps VF\_Port 1/2/26 to N\_Port 2/0/4.

```
device# configure terminal
device(config)# rbridge-id 2
device(config-rbridge-id-2)# ag
device(config-rbridge-id-2-ag)# nport interface fiberchannel 2/0/4
device(config-rbridge-id-2-ag-nport-if-fi-2/0/4)# map fport interface fcoe 1/2/26
```

## History

Release version	Command history
6.0.1	This command is available only as an independent command, in AG configuration mode. Previously, in RBridge-ID configuration mode it could be executed with the <b>ag</b> prefix.

## Related Commands

[ag](#), [nport interface fibrechannel \(port group\)](#), [show ag](#)

# nport interface fibrechannel (port group)

Under Access Gateway, adds or deletes N\_Ports from a port group.

## Syntax

```
nport interface fibrechannel port
```

```
no nport interface fibrechannel port
```

## Command Default

Ports not added to a specified port group remain in the default port group (pg 0).

## Parameters

*port*

Specifies the N\_Port number. N\_Ports are identified by *rbridge-id/slot/N\_Port*, such as 3/0/4 for RBridge 3, slot 0, and N\_Port 4.

## Modes

Port Group configuration mode

## Usage Guidelines

Before adding an N\_Port to a port group, you must remove the N\_Port from its current port group—unless the port resides in default port group 0 (pg 0).

To remove an N\_Port from a port group, use the **no** form of this command.

## Examples

From Port Group configuration mode, the following example adds N\_Port 3/0/3 to port group 2 (pg 2).

```
device# configure terminal
device(config)# rbridge 3
device(config-rbridge-id-3)# ag
device(config-rbridge-id-3-ag)# pg 2
device(config-rbridge-id-3-ag-pg-2)# nport interface fibrechannel 3/0/3
```

The following example removes N\_Port 3/0/3 from port group 3 (pg 3).

```
device# configure terminal
device(config)# rbridge 3
device(config-rbridge-id-3)# ag
device(config-rbridge-id-3-ag)# pg 3
sw0(config-rbridge-id-3-ag-pg-3)# no nport interface fibrechannel 3/0/3
```

# nsx-controller client-cert

Generates a self-signed certificate for VXLAN gateway NSX Controller connections or deletes a previously generated certificate.

## Syntax

```
nsx-controller client-cert generate digest { sha1 | sha256 | sha512 }
```

```
nsx-controller client-cert delete
```

## Command Default

If a digest algorithm is not specified, the SHA-1 algorithm is used.

## Parameters

### generate digest

Generates a self-signed certificate for the VXLAN gateway.

#### sha1

Specifies the SHA-1 digest algorithm for signing the certificate.

#### sha256

Specifies the SHA-256 digest algorithm for signing the certificate.

#### sha512

Specifies the SHA-512 digest algorithm for signing the certificate.

### delete

Deletes a previously generated certificate.

## Modes

Privileged EXEC mode

## Usage Guidelines

### ATTENTION

The SHA-2 family of algorithms is supported only for Open vSwitch Database (OVSDB) connections.

If a digest algorithm is not specified, the SHA-1 algorithm is used.

Only one certificate can be generated. Additional attempts will result in an error.

Use the **show nsx-controller client-cert** command to view the generated certificate.

During an upgrade from NOS 6.x.x or NOS 7.0.x, the existing (SHA-1) certificate is retained. To use an SHA-2 family certificate for an OVSDB connection, the previous certificate must be deleted and either an SHA-256 or SHA-512 certificate must be generated. An SHA-2 family certificate is retained following a downgrade to a release prior to NOS 7.1.0.

## Examples

The following example generates a self-signed certificate for the VXLAN gateway that uses an SHA-1 hash algorithm (the default).

```
device# nsx-controller client-cert generate digest
```

The following example generates a self-signed certificate for the VXLAN gateway that uses the SHA-512 hash algorithm for an OVSDB connection.

```
device# nsx-controller client-cert generate digest sha512
```

The following example deletes a previously signed certificate.

```
device# nsx-controller client-cert delete
```

## History

Release version	Command history
7.1.0	This command was modified to include additional digest algorithms and usage guidelines.

## Related Commands

[show nsx-controller](#)

# nsx-controller name

Creates an NSX controller connection profile or enters NSX controller configuration mode for an existing NSX controller connection profile.

## Syntax

`nsx-controller name`

`no nsx-controller name`

## Parameters

*name*

Specifies a name for the NSX controller. The name is an alphanumeric, 32-character-maximum string that can also contain hyphens and underscores.

## Modes

Global configuration mode

## Usage Guidelines

Only one NSX Controller connection profile can be configured.

The **no** form of the command deletes an NSX controller connection profile. All active connections are closed, and all tunnels related to this NSX controller are deleted.

By default, a connection profile is inactive. To activate a profile, run the **activate** command in NSX controller configuration mode.

## Examples

The following example creates an NSX controller profile named profile1.

```
device# configure terminal
device(config)# nsx-controller profile1
```

# ntp authentication-key

Creates an authentication key to associate with the NTP server, thereby enabling NTP authentication.

## Syntax

```
ntp authentication-key key-id {md5 md5-string | sha1 sha1-string}encryption-level enc_value  
no ntp authentication-key key-id
```

## Command Default

NTP authentication is disabled by default.

## Parameters

*key-id*

Specifies an ID for an authentication key. The range is from 1 through 65535.

**md5** *md5-string*

Specifies a string for the MD5 message-digest algorithm. The string can be a maximum of 15 ASCII characters.

**encryption-level** *enc\_value*

Defines the level of encryption for the NTP authentication key. The valid values are 0 and 7. The value 0 is clear text format and the value 7 is fully encrypted format. The default value is 7.

**sha1** *sha1-string*

Specifies a string for SHA1 encryption. The string can be a maximum of 15 ASCII characters.

## Modes

Global configuration mode

## Usage Guidelines

This command adds an NTP authentication key to a list of authentication keys in the database. The key is shared by the client (switch) and an external NTP server.

The maximum number of configurable NTP authentication keys is five. You cannot configure a duplicate key ID with a different key string. Use the **no ntp authentication-key** *key-id* command to remove the specified authentication key.

Authentication key must be created before associating the key with any server. Refer to the **ntp server** command for information on how to create this association.

Before downgrading the firmware to a version that does not support the encryption-level option, the encryption-level should be set to 0.



## Examples

To create an authentication key with an ID of 33, an MD5 string called *check*, and an encryption level of 0 :

```
switch# configure
switch(config)# ntp authentication-key 33 md5 check encryption-level 0
```

## Related Commands

[ntp server](#), [ntp source-ip](#)

# ntp server

Specifies or adds an NTP server IP address and optionally associates an authentication key to the server.

## Syntax

```
ntp server ip-address [ key key-id ] [ use-vrf vrf-name ]
no ntp server ip-address [ key key-id ] [ use-vrf vrf-name ]
```

## Command Default

The NTP server list is LOCL (no NTP server configured).

## Parameters

*ip-address*

Specifies the NTP server IPv4 IP address (dot-decimal notation) or the IPv6 IP address (hexadecimal colon-separated notation).

**key** *key-id*

Associates a key from the key list to the specified server. The range for a key ID is from 1 through 65535.

**use-vrf** *vrf-name*

Specifies a VRF through which to communicate with the NTP server. See the Usage Guidelines.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to add an NTP server IPv4 or IPv6 address to a list of server IP addresses, or to associate an existing authentication key with an NTP server IP address.

The maximum number of NTP servers allowed is five.

Network Time Protocol (NTP) commands must be configured on each individual switch.

Use the **no ntp server** *ip-address* command to remove the specified NTP server IP address. Removing the current active NTP server resets the NTPstatus to "LOCL" until a new, active server is selected.

Use the **no ntp server** *ip-address* **key** *key-id* command to remove the key from the specified NTP IP address.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

To associate a configured key ID of 15 to an NTP server on the management VRF:

```
device(config)# ntp server 192.168.10.1 key 15
```

To associate a configured key ID of 15 to an NTP server on a user-specified VRF:

```
device(config)# ntp server 192.168.10.1 key 15 use-vrf myvrf
```

To remove an NTP server from the current list of NTP servers on the management VRF:

```
device(config)# no ntp server 192.168.10.1
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.

# ntp source-ip

Configures the source IP address to be used to access the NTP server.

## Syntax

```
ntp source-ip [ chassis-ip ip_address | mm-ip ip_address ]
no ntp source-ip
```

## Command Default

The NTP source IP is not configured.

## Parameters

**chassis-ip** *ip\_address*

Uses the IP address of the chassis for the NTP server.

**mm-ip** *ip\_address*

Uses the management module (MM) IP address for the NTP server.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no ntp source-ip** command to remove the configuration.

## Examples

Typical command example:

```
switch# configure terminal
switch(config)# ntp source-ip chassis-ip 10.28.52.26
```

Typical command example:

```
switch# configure terminal
switch(config)# ntp source-ip mm-ip 10.28.52.27
```

## History

Release version	Command history
5.0.2	This command was introduced.

## Related Commands

[ntp server](#), [ntp authentication-key](#), [show ntp status](#)

# openflow-controller

Configures an OpenFlow controller in active connection mode.

## Syntax

**openflow-controller** *controller-name*

**openflow-controller** *ip address* [ **method** *method-name* { **no-ssl** | **ssl** } ] [ **port** *port-number* ]

**no openflow-controller** *controller-name*

## Command Default

See the Usage Guidelines.

## Parameters

*controller-name*

Specifies the user-given name for the controller.

**ip address**

Specifies the IPv4 address of the controller.

**method**

Specifies the method to connect to the OpenFlow controller.

**ssl**

Specifies an SSL connection.

**no-ssl**

Specifies a TCP connection.

**port** *port-number*

Specifies the OpenFlow controller TCP port number. Range is from 1 through 65535.

## Modes

Global configuration mode

## Usage Guidelines

Currently, only SSL is supported for the connection method. The TCP port is usually limited to either 6633 or 6653 and must match the port configured on the controller. The default port is 6633.

Use the **no** form of the command to remove the specified OpenFlow controller. You cannot remove an active controller.

See the **openflow logical-instance** command for configuring a passive controller connection.

## Examples

This example creates an OpenFlow controller and assigns an IPv4 address, method, and port.

```
device(config)# openflow-controller mycontroller
device(config-openflow-controller-mycontroller)# ip address 10.24.82.10 method no-ssl port 6633
device(config-openflow-controller-mycontroller)#
```

## History

Release version	Command history
6.0.1	This command was introduced.

# openflow enable

Enables the OpenFlow mode on an interface.

## Syntax

```
openflow enable [ layer2 ][ layer3]
```

```
{no} openflow enable [ layer2 ][ layer3]
```

## Command Default

See the Usage Guidelines.

## Parameters

**layer2**

Matches only on Layer 2 packet headers.

**layer3**

Matches only on Layer 3 packet headers.

## Modes

Interface configuration mode

## Usage Guidelines

LLDP must be explicitly disabled, by means of the **lldp disable** command, to enable OpenFlow on an interface.

Matching on Layer 2 packet headers is the default behavior.

An OpenFlow logical instance must be created, by means of the **openflow logical-instance 1** command in RBridge ID configuration mode. Then it must be associated with an interface by means of the same command in interface subtype configuration mode.



## Examples

To disable LLDP, enter the **lldp disable** command in interface subtype configuration mode.

```
device(config)#interface tengigabitethernet 12/0/12
device(conf-if-te-12/0/12)# lldp disable
```

To associate the interface to logical instance , enter the following command.

```
device(conf-if-te-12/0/12)# openflow logical-instance 1
```

To enable OpenFlow on the interface and match on Layer 2 packet headers, enter the following command.

```
device(conf-if-te-12/0/12)# openflow enable
```

To match on Layer 3 packet headers, enter the following command.

```
device(conf-if-te-12/0/12)# openflow enable Layer3
```

## History

Release version	Command history
6.0.1	This command was introduced.

# openflow logical-instance

Creates an OpenFlow logical instance, enables a variety of options under OpenFlow logical-instance configuration mode, and also associates the logical instance with an interface.

## Syntax

```
openflow logical-instance number
```

```
{no} openflow logical-instance number
```

## Command Default

This feature is disabled.

## Parameters

*number*

Specifies the logical instance number. See the Usage Guidelines.

## Modes

RBridge ID configuration mode

Interface subtype configuration mode

## Usage Guidelines

An OpenFlow logical instance is first created by means of this command in RBridge ID configuration mode, where parameters can be set. Then this same command is used in interface subtype configuration mode to associate the logical instance with the interface.

You can associate a specific controller to a logical instance, though many controllers can be configured at the global level.

Use the **no** form of the command to remove the specified OpenFlow logical instance.

The initial release supports only logical instance 1. The following commands are available under OpenFlow logical-instance configuration mode, under RBridge ID configuration mode:

<b>activate (OpenFlow)</b>	Activates the OpenFlow logical instance.
<b>controller</b>	Specifies the name of an OpenFlow controller
<b>default-behavior</b>	Configures the default behavior for a table miss.
<b>passive</b>	Configures a passive controller connection.
<b>version</b>	Specifies the OpenFlow version (currently only v1.3, the default).

Refer to the above commands for details.

## Examples

The following example specifies OpenFlow logical-instance 1 and enters OpenFlow logical-instance configuration mode.

```
device(config)# rbridge-id 12
device(config-rbridge-id-12)# openflow logical-instance 1
sw0(config-logical-instance-1)# ?
Possible completions:
  activate      Activate this logical instance
  controller    OpenFlow controller name
  default-behavior Default MISS behavior for this logical instance
  describe      Display transparent command information
  do            Run an operational-mode command
  exit         Exit from current mode
  help         Provide help information
  no          Negate a command or set its defaults
  passive     Passive controller connection
  pwd        Display current mode path
  top       Exit to top level and optionally run command
  version   OpenFlow version
```

The following example assigns the OpenFlow logical instance to an Ethernet interface, disables LLDP, and enables OpenFlow on the interface.

```
device(config)# interface tengigabitethernet 12/0/12
device(conf-if-te-12/0/12)# lldp disable
device(conf-if-te-12/0/12)# openflow logical-instance 1
device(conf-if-te-12/0/12)# openflow enable
```

## History

Release version	Command history
6.0.1	This command was introduced.

# oscmd

Provides a command shell for selected Linux commands.

## Syntax

**oscmd** *Linuxcommand*

## Parameters

*Linuxcommand*

The following Linux commands are supported with **oscmd** :

**arp** [ -a ]

Displays the Address Resolution Protocol (ARP) tables.

**cat**

Concatenates files and displays to standard output.

**cp**

Copies files and directories in a file system.

**ftp**

Transfers files to and from a remote server.

**ifconfig** [ *netmask* ] [ **up** ]

Configures the active network interface.

**ls** [ -al ] [ *path* ]

Lists files and directories on the switch.

**mkdir** *dir*

Creates a directory.

**mv** [ *i* ] *file1 file2*

Renames a file or directory.

**rm** [ -rf ] *file*

Removes a file or directory.

**rmdir**

Removes a directory.

**tcpdump**

Analyzes network traffic. The following parameters are supported with the Network OS implementation. Refer to the Linux documentation for more information on how to use this command. -

**AbdDefIKILnNOpqRStuUvxX**

- **-B** *buffer\_size*
- **-c** *count*
- **-C** *file\_size*
- **-G** *rotate\_seconds*
- **-F** *file*
- **-i** *interface*

- **-m** *module*
- **-M** *secret*
- **-r** *file*
- **-s** *snapplen*
- **-T** *type*
- **-w** *file*
- **-W** *filecount*
- **-E** *spi@ipaddr*
- **-y** *datalinktype*
- **-z** *postrotate-command*
- **-Z** *user [ expression ]*

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to execute selected Linux commands on the switch. Refer to the Linux man pages for more information on the supported commands.

The **oscmd** command is disabled under FIPS mode.

## Examples

To display the ARP table:

```
switch# oscmd arp -a
? (127.2.1.9) at ac:de:48:02:09:07 [ether] on eth2
? (127.2.1.7) at ac:de:48:02:07:07 [ether] on eth2
? (10.17.16.3) at 00:1b:ed:0b:90:00 [ether] on eth0
? (10.17.16.1) at 02:e0:52:5a:36:5c [ether] on eth0
? (10.17.19.14) at 00:14:22:20:5c:3c [ether] on eth0
? (127.2.2.9) at ac:de:48:02:09:08 [ether] on eth2
```

To copy a file to a remote server:

```
switch# oscmd rcp file root@127.2.1.8:
switch#
switch:FID128:root# telnet 127.2.1.8

Trying 127.2.1.8...
Connected to 127.2.1.8.
Escape character is '^]'.
Linux 2.6.34.6 (sw0) (0)
sw0 login: root

Password:
sw0:L2/0: >ls

.profile .rhosts file
```

To copy a file using secure copy:

```
switch# oscmd scp file file1 hegdes@10.31.2.27:  
hegdes@10.31.2.27's password: file  
100% 0 0.0KB/s 00:00 file1  
100% 0 0.0KB/s 00:00
```

# overlay-gateway

Creates a VXLAN overlay gateway instance and enables VXLAN overlay gateway configuration mode.

## Syntax

**overlay-gateway** *name*

**no overlay-gateway** *name*

## Command Default

The default VXLAN overlay gateway setting for **type** is **hardware-vtep**.

## Parameters

*name*

Specifies a name for the VXLAN overlay gateway. Only one gateway instance can be configured. The name is an alphanumeric, 32-character-maximum string that can also contain hyphens and underscores.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to create a VXLAN overlay gateway instance with the given name. An overlay network is a virtual network that is built on top of existing network Layer 2 and Layer 3 technologies. The objectives of setting up a gateway are:

- Configuring the source IP address
- Configuring the VLAN or VLANs
- Configuring MAC addresses to export to the VXLAN domain
- Enabling statistics collection for VLAN domains
- Enabling SPAN.

Once you create the gateway instance, you enter VXLAN overlay gateway configuration mode, where you can configure other properties for this gateway. The key commands available in this mode are summarized below:

**TABLE 11** Key commands available in VXLAN overlay gateway configuration mode

Command	Description
<b>activate</b>	Activates a VXLAN overlay gateway instance.
<b>attach rbridge-id</b>	Assigns a range of RBridge IDs to a VXLAN overlay gateway instance.
<b>attach vlan</b>	Specifies exported VLANs or MAC addresses in VXLAN overlay gateway configurations
<b>enable statistics direction</b>	Enables per-VLAN statistics collection for a VXLAN overlay gateway instance.
<b>ip access-group</b>	Sets an IPv4 ACL for the gateway.

**TABLE 11** Key commands available in VXLAN overlay gateway configuration mode (continued)

Command	Description
<b>ip interface loopback</b>	Sets the loopback port number for the overlay gateway instance.
<b>ip interface Ve</b>	Sets the IP address of a VXLAN overlay gateway instance.
<b>ipv6 access-group</b>	Sets an IPv6 ACL for the gateway.
<b>mac access-group</b>	Sets a MAC ACL for the gateway.
<b>map vlan vni</b>	In a VXLAN overlay gateway configuration that uses Layer 2 extension, associates VLANs with VXLAN Network Identifiers (VNIs).
<b>monitor session</b>	Enables switched port analyzer (SPAN) on one or all tunnels of a VXLAN gateway.
<b>sflow</b>	Enables sFlow monitoring of the tunnel endpoints for a VXLAN overlay gateway.
<b>site</b>	Configures a remote Layer 2 extension site in a VXLAN overlay gateway context.
<b>type</b>	Specifies whether a VXLAN overlay gateway uses NSX Controller/ OpenStack integration or Layer 2 extension.

Only one VXLAN overlay gateway instance can be configured.

Use the **no overlay-gateway** command to delete the VXLAN overlay gateway instance from the cluster. All tunnels for the gateway are also deleted. There are no other **no** forms of this command.

By default, a VXLAN overlay gateway instance is inactive. To activate an instance, first configure its other properties (such as which R Bridges it attaches to), and then enter the **activate** command.

Note the following conditions related to changing the VXLAN gateway type:

- Before changing **type**, ensure that no R Bridge is attached to the gateway.
- If changing **type** from **hardware-vtep** to **layer2-extension**, ensure that there are no "attach vlan" configurations.
- If changing **type** from **layer2-extension** to **hardware-vtep**, ensure that no "map vlan" configurations are present.

#### NOTE

The running configuration always shows the setting for **type**, even the default value (**hardware-vtep**). This means that when an overlay gateway is created, "type hardware-vtep" automatically appears in the running configuration.

Note the following conditions for related commands:

- The **attach vlan** command is valid only when **type** is **hardware-vtep**.
- The **map vlan vni** command is valid only when **type** is **layer2-extension**.

## Examples

To create a VXLAN overlay gateway instance named "gateway1" and enter VXLAN overlay gateway configuration mode:

```
switch# config
switch(config)# overlay-gateway gateway1
switch(config-overlay-gw-gateway1)#
```



## History

Release version	Command history
7.0.0	The <b>nsx</b> keyword was deprecated and replaced by <b>hardware-vtep</b> .

# ovsdb-server

Specifies an Open vSwitch Database SSL server for OpenStack deployments.

## Syntax

```
ovsdb-server name ]
no ovsdb-server name ]
```

## Command Default

See the Usage Guidelines.

## Parameters

*name*  
Name of an OVSDB SSL server.

## Modes

Global configuration mode.

## Usage Guidelines

**NOTE**  
Currently only one server can be configured.

The SSL server always runs in the context of the management VRF, on the principle node where the logical interface resides. Use the **no** form of this command to delete the SSL server instance and reset all its connections.

## Examples

To specify the SSL server "myserver":

```
device# ovsdb-server myserver
device(config-ovdsb-server-myserver) #
```

## History

Release version	Command history
7.0.0	This command was introduced.

# Commands P through short-path-forwarding

---

## passive

Specifies the behavior of a passive OpenFlow controller connection in OpenFlow logical-instance configuration mode.

## Syntax

**passive** *connection-type* [ **port** *port-num* ] [ **ip address** *IPv4\_address* ] [ **use-vrf** *vrf-name* ]

**no passive** *connection-type* [ **use-vrf** *vrf-name* ]

## Parameters

*connection-type*

The connection type. See the Usage Guidelines.

**port** *port-num*

Specifies a TCP port to which remote controllers connect. Range is from 1 through 65535.

**use-vrf** *vrf-name*

Specifies a VRF through which to connect to the controller. See the Usage Guidelines.

## Modes

OpenFlow logical-instance configuration mode

## Usage Guidelines

The keyword **no-ssl** is the only available option. Use the **no** form of this command to remove the passive instance.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

The following example enters OpenFlow logical-instance configuration mode, and specifies a passive connection through a user-defined VRF.

```
device(config)# rbridge-id 12
device(config-rbridge-id-12)# openflow logical-instance 1
device(config-logical-instance-1)# passive no-ssl use-vrf myvrf
device(config-logical-instance-1)#
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.

# password-attributes

Configures global password attributes.

## Syntax

```
password-attributes { admin-lockout [ character-restriction { lower numlower | numeric numdigits | special-char numsplchars
| upper numupper } | character-restriction | max-retry maxretry | min-length minlen | max-lockout-duration duration }
```

```
no password-attributes [ min-length | max-retry | character-restriction | max-lockout-duration ]
```

```
no password-attributes admin-lockout
```

## Command Default

The default for *min-length* is 8. All other defaults are 0.

## Parameters

### admin-lockout

Enables lockout for admin role accounts.

### character-restriction

Configures the restriction on various types of characters.

#### lower *numlower*

Specifies the minimum number of lowercase alphabetic characters that must occur in the password. Values range from 0 through 32 characters. The default value is 0.

#### numeric *numdigits*

Specifies the minimum number of numeric characters that must occur in the password. Values range from 0 through 32 characters. The default is 0.

#### special-char *numsplchars*

Specifies the number of punctuation characters that must occur in the password. All printable, nonalphanumeric punctuation characters, except colon (:) are allowed. Values range from 0 through 32 characters. The default value is 0.

#### upper *numupper*

Specifies the minimum number of uppercase alphabetic characters that must occur in the password. Values range from 0 through 32 characters. The default value is 0.

### max-retry *maxretry*

Specifies the number of failed password logins permitted before a user is locked out. Values range from 0 through 16 attempted logins. The default value is 0.

### min-length *minlen*

Specifies the minimum length of the password. Valid values range from 8 through 32 characters. The default is 8 characters.

### max-lockout-duration *duration*

Specifies the maximum number of minutes after which the user account is unlocked. Range is from 0 through 99999. The default is 0, representing an infinite duration.

## Modes

Global configuration mode

## Usage Guidelines

Lockout policy locks admin role accounts when the user exceeds the configured maximum number of failed login attempts.

To reset password attributes to their default values, enter the **no** form of this command.

## Examples

The following example configures global password attributes and verifies the configuration.

```
device#configure terminal
device(config)# password-attributes max-retry 4
device(config)# password-attributes character-restriction lower 2
device(config)# password-attributes character-restriction upper 1 numeric 1 special-char 1
device(config)# exit
device# show running-config password-attributes

password-attributes max-retry 4
password-attributes character-restriction upper 1
password-attributes character-restriction lower 2
password-attributes character-restriction numeric 1
password-attributes character-restriction special-char 1
```

The following example resets the character restriction attributes and verifies the configuration.

```
device#configure terminal
device(config)# no password-attributes character-restriction lower
device(config)# no password-attributes character-restriction upper
device(config)# exit
device# show running-config password-attributes

password-attributes max-retry 4
password-attributes character-restriction numeric 1
password-attributes character-restriction special-char 1
```

The following example clears all global password attributes.

```
device#configure terminal
device(config)# no password-attributes
device(config)# exit
device# show running-config password-attributes

% No entries found.
```

The following example sets the maximum number of retries to 3 and enables lockout policy for admin role accounts.

```
device#configure terminal
device(config)# password-attributes max-retry 3 admin-lockout
```

The following example specifies that the user account be unlocked after 5 minutes and enables lockout policy for admin role accounts.

```
device#configure terminal
device(config)# password-attributes max-lockout-duration 5 admin-lockout
```

## History

Release version	Command history
6.0.0	This command was consolidated and updated.
6.0.1	The <b>max-lockout-duration</b> keyword was added.

## Related Commands

[rule](#), [service password-encryption](#), [show running-config password-attributes](#)

# pdu-rx-limit

Sets the number of PDU packets received on an ELD-enabled port before detecting and breaking a loop.

## Syntax

`pdu-rx-limit limit`

`no pdu-rx-limit limit`

## Command Default

The default is 1.

## Parameters

*limit*

The number of PDU packets. The valid range is 1 through 5.

## Modes

ELD configuration mode

## Usage Guidelines

This command sets the same value for every RBridge in the cluster.

Use this command with the **hello-interval** command to determine the time taken to detect a loop. The time taken to detect a loop is the product of the pdu-rx-limit and the hello interval. The cluster in the loop with the lowest pdu-rx-limit is the cluster where the loop gets broken, assuming that the hello limit is correctly set to the same value on all RBridges.

This functionality detects Layer 2 loops only.

Enter **no pdu-rx-limit** to reset the limit to its default value.

## Examples

The following example sets the limit on the number of PDU packets received to 4.

```
device# configure terminal
device(config)# protocol edge-loop-detection
device(config-eld)# pdu-rx-limit 4
```



# permit ip host

In an ARP ACL, create a rule that permits ARP messages from a host specified by both IP and MAC addresses, which is one of the steps implementing dynamic ARP inspection (DAI) on a VLAN. You can also specify logging for such a rule.

## Syntax

```
permit ip host sender-ip mac host sender-mac-address [ log ]  
no permit ip host sender-ip mac host sender-mac-address [ log ]
```

## Command Default

No **permit** rules are defined.

## Parameters

*sender-ip*

Specifies the sender IP address.

**mac host** *sender-mac-address*

Specifies the sender MAC address, in hexadecimal format.

**log**

Enables logging for this **permit** rule. For additional requirements for logging, refer to the Usage Guidelines.

## Modes

ARP-ACL configuration mode

## Usage Guidelines

To remove the **permit** rule from the ACL, use the **no** form of this command.

On untrusted interfaces of DAI-enabled VLANs, incoming ARP packets from permitted IP/MAC addresses are accepted only if all of the following steps were performed:

- Create the ACL, using the **arp access-list** command.
- In the ACL, create one or more rules, using the **permit ip host** command. Each rule specifies an IP/MAC address-pair.
- Apply the ACL to one or more VLANs, using the **ip arp inspection filter** command.
- Enable DAI on such VLANs, using the **ip arp inspection** command.

## Examples

The following example does the following:

1. Creates an ARP ACL named "host2".
2. Defines a **permit** rule in that ACL.
3. Applies the ACL to VLAN 200.
4. Enables dynamic ARP inspection (DAI) on VLAN 200.

```
device# configure terminal
device(config)# arp access-list host2
device(config-arp-acl)# permit ip host 1.1.1.1 mac host 0000.0011.0022
device(config-arp-acl)# exit
device(conf)# interface vlan 200
device(conf-if-vlan-200)# ip arp inspection filter host2
device(conf-if-vlan-200)# ip arp inspection
```

The following example creates a **permit** rule within the arp access-list command.

```
device# configure terminal
device(config)# arp access-list host2 permit ip host 1.1.1.1 mac host 0000.0011.0022
```

## History

Release version	Command history
6.0.1	This command was introduced.

## pg

Under Access Gateway configuration mode, creates or accesses an N\_Port group.

## Syntax

```
pg pgid
no pg pg_id
```

## Parameters

*pg\_id*  
Specifies the numerical port group identifier. Valid values are 1 through 15. (The value of the default port group is 0.)

## Modes

Access Gateway (AG) configuration mode

## Usage Guidelines

This command configures a port group with a unique ID (*pg\_id*). Once configured, you can access the port group in Port Grouping configuration mode to perform configuration tasks, such as adding and removing N\_Ports, enabling port group modes, and renaming the group.

To remove a port group, use the **no** form of this command.

## Examples

The following command creates port group 1 and enables Port Grouping configuration mode for the port group.

```
device# configure terminal
device(config)# rbridge 3
device(config-rbridge-id-3)# ag
device(config-rbridge-id-3-ag)# pg 1
device(config-rbridge-id-3-ag-pg-1)#
```

The following command removes port group 1.

```
device# configure terminal
device(config)# rbridge 3
device(config-rbridge-id-3-ag)# no pg 1
```

## History

Release version	Command history
6.0.1	This command is available only as an independent command, in AG configuration mode. Previously, in RBridge-ID configuration mode it could be executed with the <b>ag</b> prefix.

## Related Commands

ag, map fport interface fcoe, show ag pg

# ping

Verifies network connectivity between a source and a destination on a TCP/IP network.

## Syntax

```
ping dest-IPv4_addr [ ipv6 dest-ipv6-addr ] [ host-name ] [ count [ number ] [ interface { <N> gigabitethernet rbridge-id/slot/
port | management | ve vlan_id } ] ] [ timeout seconds ] [ datagram-size bytes ] [ quiet ] [ numeric ] [ vrf vrf-name ]
```

## Command Default

The default for count is 5. The default for timeout is 1. The default for datagram-size is 56.

## Parameters

*dest-IPv4\_addr*

Specifies the IPv4 address of the destination device.

**ipv6** *dest-ipv6-addr*

Specifies the IPv6 address of the destination device.

*host-name*

Destination host name. The default value is 1.

**count** *number*

Specifies the number of transmissions (pings). The range is from 1 through 7200.

**interface** <N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port. The interface option is only available for the **ipv6 link-local address ping** command.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**interface management**

Specifies the management interface.

**interface ve** *vlan\_id*

Specifies the interface is a virtual Ethernet, and specifies the VLAN ID of the interface.

**timeout** *seconds*

Specifies the time (in seconds) to wait for a response. The range is from 1 through 60. The default value is 1.

### NOTE

This option applies only to IPv4.

**datagram-size** *bytes*

Specifies the datagram size (also known as the maximum transmission unit, or MTU) in bytes. The range is from 36 through 9100. The default value is 56.

**quiet**

Prints only the first and last line of the command output.

**numeric**

Does not lookup hostnames.

**vrf** *vrf-name*

Pings the specified VRF instance. If no VRF is specified, the default-vrf is pinged. See the Usage Guidelines.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command sends a specified number of pings with configured parameters to the specified destination device.

To ping management routes, use the **ping vrf** or **ping ipv6 vrf** command and enter **mgmt-vrf** as follows. You must enter the name of the management VRF manually.

```
device# ping vrf mgmt-vrf
device# ping ipv6 vrf mgmt-vrf
```

## Examples

To ping an IPv4 destination address:

```
device# ping 172.16.4.80
Type Control-c to abort
PING 172.16.4.80 (172.16.4.80): 56 data bytes
64 bytes from 172.16.4.80: icmp_seq=0 ttl=120 time=101.466 ms
64 bytes from 172.16.4.80: icmp_seq=1 ttl=120 time=122.914 ms
64 bytes from 172.16.4.80: icmp_seq=2 ttl=120 time=145.637 ms
64 bytes from 172.16.4.80: icmp_seq=3 ttl=120 time=170.032 ms
64 bytes from 172.16.4.80: icmp_seq=4 ttl=120 time=103.036 ms
--- 172.16.4.80 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 101.466/128.617/170.032/26.188 ms
```

To ping an IPv4 destination address in quiet mode:

```
device# ping 172.16.4.80 quiet
Type Control-c to abort
PING 172.16.4.80 (172.16.4.80): 56 data bytes
--- 172.16.4.80 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 100.605/146.372/192.552/32.505 ms
```

To ping an IPv6 destination address in numeric mode with a datagram size:

```
device# ping ipv6 fec0:60:69bc:92:218:8bff:fe40:1470 count 3 datagram-size 48
numeric timeout 3
Type Control-c to abort
PING fec0:60:69bc:92:218:8bff:fe40:1470 (fec0:60:69bc:92:218:8bff:fe40:1470): 48
data bytes
56 bytes from fec0:60:69bc:92:218:8bff:fe40:1470: icmp_seq=0 ttl=64 time=6.356 ms
56 bytes from fec0:60:69bc:92:218:8bff:fe40:1470: icmp_seq=1 ttl=64 time=0.170 ms
56 bytes from fec0:60:69bc:92:218:8bff:fe40:1470: icmp_seq=2 ttl=64 time=0.171 ms
--- fec0:60:69bc:92:218:8bff:fe40:1470 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.170/2.232/6.356/2.916 ms
```

To ping management routes, use the **ping vrf** or **ping ipv6 vrf** command and enter **mgmt-vrf** as follows. You must enter the name of the management VRF manually.

```
device# ping vrf mgmt-vrf

device# ping ipv6 vrf mgmt-vrf
```

## Related Commands

[traceroute](#)

# police cir

Sets the committed information rate for a class-map.

## Syntax

```
police cir cir-rate
```

```
no police cir
```

## Parameters

*cir-rate*

Committed information rate. Valid values range from 40000 through 40000000000 bps in multiples of 40000.

## Modes

Policy-map class configuration mode

## Usage Guidelines

When you are in config-policy-map-class mode launching the **police cir cir-rate** command places the system in config-policy-map-class-police mode for the configured class-map. At this point, you can add or remove additional policing parameters for the class-map.

Only the **police cir** and **cbs** commands are mandatory for configuring a class-map.

If the optional parameters for a class-map are not set, they are treated as disabled. To delete parameters for a class-map, you must delete all policer parameters while in the policy-map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

Use the **no** version of this command to remove the parameter from the class-map.

## Examples

This example configures a class map called "default" within a policy map.

```
device# configure terminal
device(config)# policy-map policymap1
device(config-policymap)# class default
device(config-policymap-class)# police cir 40000
device(config-policymap-class)#
```



# police-priority-map

Creates color-based priority CoS mapping. A police-priority-map remaps frame CoS values to conform or exceed color values when rates conform or exceed limits set in a class map.

## Syntax

**police-priority-map** *name*

**no police-priority-map** *name*

**conform** *CoSvalues*

**exceed** *CoSvalues*

## Command Default

If you do not define priority mapping for a color (conform or exceed), the map defaults to priorities 0, 1, 2, 3, 4, 5, 6, and 7.

## Parameters

*name*

Name of police-priority map

*CoSvalues*

CoS priority values (0, 1, 2, 3, 4, 5, 6, 7)

## Modes

Global configuration mode

Police-priority-map configuration mode

## Usage Guidelines

This command creates a police priority map.

When you launch the **police-priority-map** command, the system is placed in config-policepmap mode for the configured map. At this point, you can remap CoS values to conform or exceed color values.

Enter **conform** *CoSvalues* or **exceed** *CoSvalues* while in config-policepmap mode to remap 802.1p CoS values that are conforming to CIR values set in the policy map or exceeding CIR values, but conforming to EIR values set in the policy map.

Enter **no police-priority-map** *name* while in global configuration mode to remove the police priority map.

Enter **no conform** command or the **exceed** *CoSvalues* while in config-policepmap mode to remove CoS remapping.

### NOTE

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

## Examples

To create a priority-map and place system into config-policepmap mode to configure conform and exceed color mapping:

```
device# configure terminal
device(config)# police-priority-map pmap1
device(config-policepmap)# conform 0 1 1 2 1 2 1 1
device(config-policepmap)# exceed 3 3 3 3 4 5 6 7
```

To remove the conform class mapping while in config-policepmap mode:

```
device# configure terminal
device(config)# police-priority-map pmap1
device(config-policepmap)# no conform
```

To remove the class-map while in global configuration mode:

```
device# configure terminal
device(config)# no police-priority-map pmap1
```

## Related Commands

[show running-config police-priority-map](#)

# policy

Creates and modifies user-defined policies for Monitoring and Alerting Priority Suite (MAPS).

## Syntax

`policy policy_name`

`no policy policy_name`

## Command Default

No user-defined policies are created.

## Parameters

*policy\_name*

The name of the user-defined policy.

## Modes

MAPS configuration mode.

## Usage Guidelines

Use the `no policy` command to delete the policy.

This command creates the user-defined MAPS policy. After creation, the user defined rules and groups are applied.

When a rule is added to a policy it is associated with one or more actions. These actions are triggered by MAPS when a rule is evaluated to true.

## Examples

Typical command example.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# maps
device(config-rbridge-id-1-maps)# policy TempMonitor
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[rule \(MAPS\)](#), [group](#)

# policy-map

Configures a policy map containing a class map so that you can apply policer and QoS attributes to a particular interface.

## Syntax

```
policy-map policy-mapname
```

```
no policy-map policy-mapname
```

## Command Default

No policy map is created.

## Parameters

*policy-mapname*

Name of police policy map

## Modes

Global configuration mode

## Usage Guidelines

When you launch the **policy-map** command, the system is placed in `config-policymap mode` for the configured map. At this point, you can add a class map containing policing parameters to the policy map. (Refer to the description of the **class** command.)

This command creates a policer policy map to apply policer and QoS attributes to a particular interface. Each policy map can contain up to 32 class maps. The class map can be associated with specific policing and QoS parameters.

Maximum number of policy map creations are 128

Associate the policy map to the interface for inbound or outbound direction with the **service-policy** command.

Enter **no policy-map** *policy-mapname* while in global configuration mode to remove the policy map.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

## Examples

Create a policy map and place system into config-policymap mode so that you can add a class map.

```
device# configure terminal
device(config)# policy-map policymap1
device(config-policymap)#
```

Remove the policy map while in global configuration mode.

```
device# configure terminal
device(config)# no policy-map policymap1
```

# port (OVSDB)

Specifies the port of an Open vSwitch Database SSL server to be used for OpenStack deployments.

## Syntax

```
port port-number ]
no port port-number ]
```

## Command Default

See the Parameters.

## Parameters

*port-number*  
Specifies the port number. The default port is 6640. Range is .

## Modes

OVSDB server configuration mode

## Usage Guidelines

Use the **no** form of this command to revert to the default port.

### ATTENTION

You must stop the server before you can execute the **no** form.

## Examples

To change the SSL server port to 8000:

```
device# ovdsb-server myserver
device(config-ovdsb-server-myserver)# port 8000
```

## History

Release version	Command history
7.0.0	This command was introduced.

# port-channel

Adds the port-channel as a member of a port-channel redundancy group.

## Syntax

```
port-channel po-id [ active ]
```

```
no port-channel po-id
```

## Modes

Port-channel-redundancy-group configuration mode

## Usage Guidelines

Port-channel redundancy groups must have two port-channels as members and these port-channels can be from the same or different RBridges of a cluster.

You can specify which port-channel become active port-channel when this group gets activated. This is optional, if user doesn't provided active member, system will automatically select port-channel which comes up first as active.

The **no port-channel** command deletes the designated group.

## Examples

Typical command execution example:

```
device# configure terminal
device(config)# port-channel-redundancy-group 32
device(config-port-channel-redundancy-group-32)# port-channel 3 active
```

## Related Commands

[port-channel-redundancy-group](#)

# port-channel-redundancy-group

Enables the port-channel-redundancy-group configuration mode.

## Syntax

```
port-channel-redundancy-group group-id  
no port-channel-redundancy-group group-id
```

## Modes

Global configuration mode

## Usage Guidelines

In this configuration mode, the **port-channel** command can add port-channels as members to the port-channel redundancy-group, and specify which port-channel becomes the active port-channel when the group is activated .

A port-channel redundancy group must have two port-channels as members. These port-channels can be from the same or different R Bridges in a cluster.

The **no port-channel-redundancy-group** command deletes the designated group.

## Examples

Typical command execution example:

```
device# configure terminal  
device(config)#port-channel-redundancy-group 27  
device(config-port-channel-redundancy-group-27)#
```

## Related Commands

[port-channel](#)

# port-channel path-cost

Sets the port channel path cost behavior.

## Syntax

```
port-channel path-cost [ custom | standard ]
```

## Command Default

Path cost is standard.

## Parameters

### custom

Specifies to use the custom behavior, which sets the path cost changes according to the port-channel's bandwidth.

### standard

Specifies to use the standard behavior, which sets that the path cost does not change according to port-channel's bandwidth.

## Modes

Spanning tree configuration mode

## Usage Guidelines

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes.

## Examples

To set the behavior for the path cost to custom:

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# port-channel path-cost custom
```

```
device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# port-channel path-cost custom
```

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# port-channel path-cost custom
```

```
device# configure terminal
device(config)# protocol spanning-tree pvst
device(conf-pvst)# port-channel path-cost custom
```

```
device# configure terminal
device(config)# protocol spanning-tree rpvst
device(conf-rpvst)# port-channel path-cost custom
```



To set the behavior for the path cost to standard:

```
device# configure terminal
device(config)# protocol spanning-tree stp
device(conf-stp)# port-channel path-cost standard
```

```
device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# port-channel path-cost standard
```

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# port-channel path-cost standard
```

```
device# configure terminal
device(config)# protocol spanning-tree pvst
device(conf-pvst)# port-channel path-cost standard
```

```
device# configure terminal
device(config)# protocol spanning-tree rpvst
device(conf-rpvst)# port-channel path-cost standard
```

# port-group

Enables port-group configuration mode for VDX 8770 device 27x40 GbE line cards. The mode is a prerequisite reserved for configuring Performance and Density operating modes on these line cards.

## Syntax

```
port-group rbridge-id/slot/port-group-id
```

## Parameters

### rbridge-id

A unique identifier for the device. Values are from 1 through 239.

### slot

Specifies a valid slot number.

### port-group-id

A port group number (1-9) specific to the Extreme VDX 8770 device 27x40 GbE line card.

## Modes

Hardware configuration mode

## Usage Guidelines

When you launch the **port-group** command, the system is placed in configuration mode for the port group. At this point, you can configure Performance or Density operating modes for the port group.

Port groups on the 27x40 GbE line card are sequentially numbered starting with 1 for ports 1-3 and ending with 9 for ports 25-27. Refer to the *Extreme 8770-4 Hardware Reference Manual* or **8770-8 Hardware Reference Manual** "Overview" chapter for more information on these port groups and configuring operating modes for this line card.

### NOTE

This command is only supported on 27x40 GbE line cards installed on Extreme VDX 8770-4, VDX 8770-8, and later devices.

## Examples

To enable port-group configuration mode for port group 9 on a line card located in slot 3 on a device with RBridge ID 1:

```
device# configure terminal
device(config)# hardware
device(config-hardware)# port-group 1/3/9
device(config-port-group-1/3/9)
```

## Related Commands

[hardware, port-group, mode \(27x40 GbE line card\)](#)

# port-profile (global configuration mode)

Creates a new Automatic Migration of Port Profiles (AMPP) port-profile in the fabric.

## Syntax

```
port-profile profile-name [ activate | qos-profile | security-profile | vlan-profile | static mac-address ]
no port-profile profile-name
```

## Parameters

*profile-name*

A fabric-wide unique name of a port-profile.

**activate**

Activates the specified profile

**qos-profile**

Enters directly into edit mode for the QoS sub-profile.

**security-profile**

Enters directly into edit mode for the security sub-profile.

**vlan-profile**

Enters directly into edit mode for the VLAN sub-profile.

**static *mac-address***

Statically associates the profile VM MAC address.

## Modes

Global configuration mode

## Usage Guidelines

If the port-profile name already exists, the device enters port-profile mode and edits the existing profile. A system-generated fabric-wide unique port-profile ID is assigned by default.

You can directly access the submodes for the profile, and assign the profile statically to a MAC address.

Security profiles are applied to the ACLs based on the profile or PolicyID. Therefore, multiple security profiles can be applied to the same profiled port.

Enter **no port-profile *profile-name*** to de-activate the port-profile.

## Examples

Typical command execution example:

```
device# configure terminal
device(config)# port-profile vml-port-profile activate
```

# port-profile (port-profile-domain configuration mode)

Adds an Automatic Migration of Port Profiles (AMPP) port-profile into a specific domain in a Virtual Fabrics context.

## Syntax

**port-profile** *port-profile-name*

**no port-profile** *port-profile-name*

## Parameters

*port-profile-name*

A fabric-wide unique name of a port-profile. Range is from 1 through 128 ASCII characters.

## Modes

Port-profile-domain configuration mode

## Usage Guidelines

You must first issue the **port-profile-domain** command to enter port-profile-domain configuration mode.

In a Virtual Fabrics context, use the **port-profile-port** command to associate a profiled port to a single port-profile or a port-profile domain.

## Examples

Creating a port-profile in global configuration mode:

```
device# configure terminal
device(config)# port-profile PP_Tenant_A
```

Creating a VLAN profile and enabling 802.1Q VLAN access on a trunk:

```
device# configure terminal
device(config-port-profile-PP_Tenant_A)# vlan-profile
device(config-vlan-profile)# switchport mode trunk allow vlan add 10
```

In a Virtual Fabrics context, creating extended VLAN profiles (VLAN IDs > 4095) to include service or transport VFs and C-TAGs.:

```
device# configure terminal
device(config)# port-profile PP_Tenant_B
device(config-vlan-profile)# switchport mode trunk allow vlan add 5000 ctag 20
device(config-vlan-profile)# switchport mode trunk allow vlan add 6000 ctag 30
```

In a Virtual Fabrics context, adding port-profiles to a port-profile domain.

```
device# configure terminal
device(config)# port-profile-domain vCenter1
device(configport-profile-domain-vCenter1)# port-profile PP_Tenant_A
device(configport-profile-domain-vCenter1)# port-profile PP_Tenant_B
```

## Related Commands

port-profile-domain, vlan-profile (AMPP)

# port-profile-domain

Creates an Automatic Migration of Port Profiles (AMPP) port-profile domain that contains all of the port-profiles that can be applied to a profiled port in a Virtual Fabrics context.

## Syntax

**port-profile-domain** *port-profile-domain-name*

**no port-profile-domain** *port-profile-domain-name*

## Parameters

*port-profile-domain-name*

A fabric-wide unique name of a port-profile domain. The range is from 1 through 128 ASCII characters.

## Modes

Global configuration mode

## Usage Guidelines

Within this domain, a service or transport VF (VLAN ID > 4095) must not have overlapping 802.1Q classification tags.

The **no** form of this command deletes a port-profile domain.

Use the **port-profile-port** command to associate a profiled port to a port-profile domain.

## Examples

Creating a port-profile domain:

```
device# configure terminal
device(config)# port-profile-domain my_PP_domain
```

Adding profiles to the above domain:

```
device# configure terminal
device(config-port-profile-domain-my_PP_domain)# port-profile my_PP_domain_2
device(config-port-profile-domain-my_PP_domain)# port-profile my_PP_domain_3
```

## Related Commands

[port-profile](#) (global configuration mode), [vlan-profile](#) (AMPP)

# port-profile-port

Activates the Automatic Migration of Port Profiles (AMPP) port-profile configuration mode on a port.

## Syntax

```
port-profile-port [ domain port-profile-domain-name ]
no port-profile-port [ domain port-profile-domain-name ]
```

## Command Default

When the **domain** keyword is not used, the port-profiles in the default profile domain are used.

## Parameters

**domain**  
Selects a port-profile domain.

*port-profile-domain-name*  
Name of a port-profile domain.

## Modes

Interface subtype configuration mode

## Usage Guidelines

To apply multiple port-profiles to the interface, create and add the profiles to the default domain or to a user-created domain and apply it to the interface.

AMPP management allows you to associate AMPP port-profiles with VMware port groups, and provides a port-profile comparison tool to facilitate comparing port-profiles within or across fabrics for robust VM migration.

In a Virtual Fabrics context, use this command with the **domain** keyword to associate a profiled port to a port-profile domain. The result is that all service or transport VFs (VLAN ID > 4095) so specified are configured on the port.

- If multiple port-profiles are added to the default domain, use the **port-profile-port** command without the **domain** keyword.
- If multiple port-profiles are added to a user-created domain (for example, domain\_d1), use the **domain** keyword as in the following example: **port-profile-port domain domain\_d1**

When the **port-profile-port** command is issued without the **domain** keyword, the domain referred to is identified by "default." The default domain is automatically created by the system during a firmware upgrade from releases prior to Network OS release 4.1.0. When the upgrade is complete, this domain contains the set of port-profiles that were created before the upgrade.

Enter the **no port-profile-port** and **no shutdown** commands to remove the complete AMPP configuration from the selected port.

Enter **no port-profile-port domain *port-profile-domain-name*** to dissociate the profiled port from the port-profile domain.

**NOTE**

In VF-enabled mode only, the user can manage port-profiles in a default domain as in any other domain.

## Examples

The following examples illustrate activating AMPP port-profile configuration mode on a specific 10-gigabit Ethernet interface port.

To associate the default port-profile domain to an interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# port-profile-port
```

To associate a profiled port with a user-specified port-profile domain:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# port-profile-port domain vDC1
```

## Related Commands

[interface](#), [port-profile \(global configuration mode\)](#), [port-profile-domain](#), [shutdown](#)



# power-off

Deactivates a line card or Switch Fabric Module (SFM).

## Syntax

```
power-off { linecard | sfm } { m4_value | m8_value }
```

## Parameters

### linecard

Selects a line card to deactivate.

### sfm

Selects an SFM to deactivate.

### m4\_value

The slot number. If you are using an Extreme VDX 8770-4 device, the range of values is from 1 through 3.

### m8\_value

The slot number. If you are using an Extreme VDX 8770-8 device, the range of values is from 1 through 6.

## Modes

Global configuration mode

## Examples

To set the priority to 110 for the VRRP virtual group 1:

```
device# configure terminal
device(config)# power-off linecard 2
```

# power-off linecard

Powers off a line card.

## Syntax

```
power-off linecard slot_number
```

## Parameters

*slot\_number*

Specifies the slot number to be powered-off. Line card slots are 1 through 4 on an Extreme VDX 8770-4 and 1 through 8 on an Extreme VDX 8770-8

## Modes

Privileged EXEC mode

## Usage Guidelines

A line card must be powered off before you can change the slot configuration.

## Examples

To power off a line card in slot 4:

```
device# power-off linecard 4
```

## Related Commands

[linecard](#), [power-on linecard](#), [show running-config rbridge-id linecard](#)

## power-on

Activates a line card or device Fabric Module (SFM).

### Syntax

```
power-on { linecard | sfm } { m4_value | m8_value }
```

### Parameters

#### linecard

Selects a line card to activate

#### sfm

Selects an SFM to activate.

#### m4\_value

The slot number. If you are using an Extreme VDX 8770-4 device, the range of values is from 1 through 3.

#### m8\_value

The slot number. If you are using an Extreme VDX 8770-8 device, the range of values is from 1 through 6.

### Modes

Global configuration mode

### Examples

To set the priority to 110 for the VRRP virtual group 1:

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp
device(config-rbridge-id-101)# int te 101/1/6
device(config-if-te-101/1/6)# vrrp-group 1
device(config-vrrp-group-1)# priority 110
```

To set the priority to 110 for the VRRP-E virtual group 1:

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp-extended
device(config-rbridge-id-101)# int ve 25
device(config-ve-25)# vrrp-group-extended 1
device(config-vrrp-extended-group-1)# priority 110
```

# power-on linecard

Powers on a line card.

## Syntax

`power-on linecard slot_number`

## Parameters

*slot\_number*

Specifies the slot number to be powered-on. Line card slots are 1 through 4 on an Extreme VDX 8770-4 and 1 through 8 on an Extreme VDX 8770-8.

## Modes

Privileged EXEC mode

## Examples

To power on a line card in slot 4:

```
device# power-on linecard 4
```

## Related Commands

[linecard](#), [power-off linecard](#), [show running-config rbridge-id linecard](#)

# precedence

Sets the precedence of the CEE map.

## Syntax

**precedence** *value*

## Command Default

The default is 1.

## Parameters

*value*

The precedence value. Valid values range from 1 through 100.

## Modes

CEE map configuration mode

## Examples

To set the precedence to 1:

```
device(config-cee-map-default)# precedence 1
```

# preempt-mode

Enables or disables preempt mode for a VRRP or VRRP Extended (VRRP-E) router session.

## Syntax

```
preempt-mode  
no preempt-mode
```

## Command Default

Enabled for VRRP; Disabled for VRRP-E.

## Modes

Virtual-router-group configuration mode  
Virtual-router-extended-group configuration mode

## Usage Guidelines

This command is for VRRP and VRRP-E.

For VRRP-E, the interface must be a virtual interface (Ve).

When set, the highest-priority backup router will always be the master if the owner is not available. If not set, a higher priority backup will not preempt a lower-priority master.

Enter **no preempt-mode** to turn off preempt mode.

## Examples

To turn on preempt mode for a virtual-router-group 1 session:

```
device# configure terminal  
device(config)# rbridge-id 101  
device(config-rbridge-id-101)# protocol vrrp  
device(config-rbridge-id-101)# int te 101/1/6  
device(config-if-te-101/1/6)# vrrp-group 1  
device(config-vrrp-group-1)# preempt-mode
```

## Related Commands

[vrrp-extended-group](#), [vrrp-group](#)

# preprovision

Creates a unique RBridge-ID-to\_WWN mapping to allow a device to automatically join a logical chassis cluster. A device that uses such a mapping to join a cluster is called a bare-metal device.

## Syntax

```
preprovision rbridge-id rbridge-id wwn wwn-id
no preprovision rbridge-id rbridge-id wwn wwn-id
no preprovision rbridge-id
```

## Parameters

**rbridge-id** *rbridge-id*  
Designates the RBridge ID to be associated with the unique WWN of the specified bare-metal device. RBridge ID range is 1-239.

**wwn** *wwn-id*  
Specifies the 64-bit address that uniquely identifies the bare-metal device to be added to the cluster.

## Modes

Global configuration mode.

## Usage Guidelines

Use this command to assign an unused RBridge ID in the cluster to a bare-metal device you are adding to the cluster. The WWN of the bare-metal device can be obtained by running the **show vcs** command on the device. When you physically connect the bare-metal device to the cluster, the device obtains its necessary configuration parameters (Rbridge-ID from the Rbridge-ID-to-WWN mapping), VCS ID, VCS mode, and GVLAN state) from its neighboring device. The bare-metal device then reboots and joins the cluster, provided that its RBridge ID and RBridge-ID-to-WWN mapping are unique in the cluster.

Use the no form of this command with an RBridge ID to remove the currently configured RBridge ID from the mapping. This is useful if you have not yet physically connected the bare-metal device to the cluster, and you decide to assign a different RBridge ID to the bare-metal device when you connect it to the cluster.

Use the **no preprovision rbridge-id** command to remove all existing preprovision entries.

Each time you plan to add a bare-metal device, you need to create a new, unique RBridge-ID-to-WWN mapping.

## Examples

The following command shows how to configure an RBridge-ID-to-WWN mapping where you want to assign the RBridge ID of 10 to the example WWN:

```
device# configure terminal
device(config)# preprovision rbridge-id 10 wwn 10:11:12:13:14:15:16:17
```

## History

Release version	Command history
6.0.0	This command was introduced.

## Related Commands

[firmware download](#), [write erase](#), [show bare-metal](#)



# priority

Sets the priority of a physical router in a VRRP router group.

## Syntax

**priority** *range*

## Command Default

The default priority is 100.

## Parameters

*range*

The priority of a physical router in a virtual router group. Higher numbers have priority over lower numbers. Valid values range from 1 to 254.

## Modes

Virtual-router-group configuration mode

Virtual-router-extended-group configuration mode

## Usage Guidelines

You can perform this command for VRRP or VRRP-E.

When set, the highest priority backup router will always be the master. (For VRRP, however, the owner is always the master if it is available.) If not set, a higher priority backup will not preempt a lower priority backup that is acting as master.

For an owner router in VRRP, the priority automatically becomes 255 if the virtual IP address of the virtual router and the real IP address of the owner are the same.

## Examples

To set the priority to 110 for the VRRP virtual group 1:

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp
device(config-rbridge-id-101)# int te 101/1/6
device(config-if-te-101/1/6)# vrrp-group 1
device(config-vrrp-group-1)# priority 110
```

To set the priority to 110 for the VRRP-E virtual group 1:

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp-extended
device(config-rbridge-id-101)# interface ve 25
device(config-ve-25)# vrrp-group-extended 1
device(config-vrrp-extended-group-1)# priority 110
```

priority

## Related Commands

[vrrp-extended-group](#), [vrrp-group](#)

# priority-group-table

Configures the bandwidth for each priority group.

## Syntax

```
priority-group-table pgid [ weight weight ] [ pfc { on | off } ]
no priority-group-table pgid
```

## Command Default

There is no default value for the weight. PFC is disabled.

## Parameters

*pgid*

Specifies the priority group ID (PGID) assigned to a priority group. Valid values range from 15.0 through 15.7 for the eight reserved Strict Priority PGIDs.

**weight** *weight*

Maps a weight to a Deficit Weighted Round Robin (DWRR) scheduler queue. This parameter is only valid for the DWRR Priority Group. The sum of all DWRR Priority Group weight values must equal 100 percent. Valid values range from 1 through 100.

**pfc**

Enables the Priority-based Flow Control (PFC) for each priority that gets mapped to the priority group.

**on**

Enables PFC.

**off**

Disables PFC.

## Modes

CEE map configuration mode

## Usage Guidelines

Enter **priority-group-table** to configure the bandwidth for each priority group, to associate a weight to a DWRR scheduler queue, and to enable the PFC.

You can define up to eight additional DWRR Priority Groups with the PGID values in the range from 0 through 7. Strict Priority Groups take priority in order from the lowest PGID value to the highest PGID value; for example, a PGID of 15.0 is a higher priority than a PGID of 15.1 and a PGID of 15.1 is higher priority than a PGID of 15.2.

Enter **no priority-group-table** *pgid* to return the priority group to the default values. For the Strict Priority Group, the PGID is still valid, but the PFC is disabled. For the DWRR Priority Group, the PGID is no longer valid and is deleted; the PGID can only be deleted when it is not bound to any Priority-to-Priority Group Table entry. The following lists the bandwidth allocation to user priority groups.

**TABLE 12** Bandwidth allocation to user priority groups

PGID	PG%	PFC	Description
0	50	Y	SAN
1	50	N	LAN

A PGID value of 15 is a special value, which allows you to configure priorities with no bandwidth limit. The priority groups of 15.0 to 15.7 are predefined in the device.

## Examples

To define the CEE map and configure the bandwidth with the priority group, use the values in [Table 12](#).

```
device# configure terminal
device(config)# cee-map test
device(conf-ceemap)# priority-group-table 0 weight 50 pfc on
device(conf-ceemap)# priority-group-table 1 weight 50
```

## Related Commands

[show qos maps](#), [show running-config cee-map](#)

# priority-tag

Toggles the priority-tagging support on a specific interface.

## Syntax

**priority-tag**

**no priority-tag**

## Command Default

The priority-tag is disabled for all supported interfaces.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command is the only method for toggling priority-tagging.

Enter **no priority-tag** to disable priority-tagging support.

## Examples

To enable priority-tagging support on a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 178/0/9
device(conf-if-te-178/0/9)# priority-tag
```

## Related Commands

[interface](#)

# private-vlan

Configures a VLAN as a private VLAN (PVLAN).

## Syntax

```
private-vlan [ isolated | community | primary ]
```

```
no private-vlan [ isolated | community | primary ]
```

## Parameters

### isolated

The PVLAN is configured as an Isolated VLAN.

### community

The PVLAN is configured as a Community VLAN.

### primary

The PVLAN is configured as a Primary VLAN.

## Modes

Interface subtype configuration mode

## Examples

Typical command example:

```
device# configure terminal
device(config)# interface vlan 10
device(conf-if-vl-100)# private-vlan community
```

## Related Commands

[private-vlan association](#)

# private-vlan association

Associates a secondary VLAN to a primary VLAN.

## Syntax

```
private-vlan association [ add vlan_id | remove vlan_id ]
```

```
no private-vlan association [ add vlan_id | remove vlan_id ]
```

## Parameters

**add** *vlan\_id*

Adds the association.

**remove** *vlan\_id*

Removes the association.

## Modes

Interface subtype configuration mode

## Examples

Typical command example:

```
device# configure terminal
device(config)# interface vlan 10
device(conf-if-vl-10)# private-vlan association add 100,200
```

Typical command example:

```
device# configure terminal
device(config)# interface vlan 10
device(conf-if-vl-10)# private-vlan association remove 100,200
```

## Related Commands

[private-vlan](#)

# profile (LLDP)

Creates an LLDP profile.

## Syntax

**profile** *name*

**no profile** *name*

## Parameters

*name*

Assigns a name to the profile. The name must be between 1 and 63 ASCII characters in length.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

When you apply an LLDP profile on an interface using the **lldp profile** command, it overrides the global configuration. If a profile is not present, then the default global profile is used until you create a valid profile. Up to 64 profiles can be created.

Enter **no profile** *name* to remove the named profile.

## Examples

The following example creates a profile named test.

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)# profile test
```

The following example deletes a profile named test:

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)# no profile test
```

## Related Commands

[lldp profile](#)



# prom-access disable

Disables access to the Boot PROM for FIPS compliance.

## Syntax

```
prom-access disable
```

## Command Default

The Boot PROM is accessible.

## Modes

Privileged EXEC mode

## Usage Guidelines

In non-FIPS compliant mode, you can access the Boot PROM by holding down the ESC key during the 4-second period when the device is booting up. In FIPS compliant state, PROM access is disabled to prevent users from net-installing firmware.

Under normal operating conditions, this command is hidden to prevent accidental use. Enter **unhide fips** with the password "fibranne" to make the command available.

### ATTENTION

Use this command only when preparing a device for FIPS compliance.



### CAUTION

Once Boot PROM access is disabled, you cannot re-enable it.

## Examples

To disable access to the Boot PROM:

```
device# unhide fips
```

```
Password: *****
```

```
device# prom-access disable
```

```
You are disabling PROM access. Do you want to continue? [yes/no] (no): yes
```

```
device# PROM access Disabled
```

# protect-mode enable

Enables protect mode.

## Syntax

**protect-mode enable**  
**no protect-mode enable**

## Modes

Privileged EXEC mode

## Usage Guidelines

In the Blade Center Chassis environment, the Advanced Management Module (AMM) controls the operation of the switch by configuring and initializing it. Protect mode of operation is a special mode which needs to be supported by both the switch and the AMM. Protect mode results in the AMM ceding control to the switch. The AMM loses its ability to perform some or all of the operations on the AMM. Once the AMM cedes control to the switch, the control can be given back to the AMM only by disabling protect mode on the switch.

Once the switch enters protect mode, AMM's requests to perform any operations are ignored until the Network Administrator permits them. This behavior is preserved through power cycles, even after it is inserted into a different bay or chassis.

Enter **no protect-mode enable** to disable this command.

## Examples

Typical command example:

```
device# protect-mode enable
```

# protected-port enable

Configures a switchport as an uplink switch protected port.

## Syntax

```
protected-port enable
no protected-port enable
```

## Command Default

Uplink switch protected port is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

For this feature to be enabled on an interface, the protect port feature must be enabled globally, by means of the **uplink-switch enable** command. The interface must be enabled as a switchport, trunk mode must be enabled, and user-specified VLANs must be added.

Use the **no** form of this command to disable this feature on an interface.

## Examples

The following example configures a switchport as an uplink switch protected port.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/1
device(conf-if-te-1/0/1)# switchport
device(conf-if-te-1/0/1)# switchport mode trunk
device(conf-if-te-1/0/1)# switchport trunk allowed vlan add 6000 ctag 10
device(conf-if-te-1/0/1)# protected-port enable
```

## History

Release version	Command history
7.2.0	This command was introduced.

# protocol edge-loop-detection

Sets the edge-loop detection (ELD) configuration mode.

## Syntax

```
protocol edge-loop-detection
```

## Command Default

ELD configuration mode is not set.

## Modes

Global configuration mode

## Usage Guidelines

This functionality detects Layer 2 loops only.

## Examples

To enter the ELD configuration mode:

```
device# configure terminal
device(config)# protocol edge-loop-detection
device(config-eld)#
```

## Related Commands

[hello-interval](#), [pdu-rx-limit](#), [shutdown-time](#)

# protocol lldp

Enters the Link Layer Discovery Protocol (LLDP) configuration mode.

## Syntax

```
protocol lldp
```

```
no protocol lldp
```

## Command Default

The LLDP and DCBX protocols are enabled.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no protocol lldp** to restore the default settings.

## Examples

To enter LLDP mode:

```
device# configure terminal
device(config)# protocol lldp
device(conf-lldp)#
```

To reset all LLDP configurations:

```
device# configure terminal
device(config)# no protocol lldp
device(conf-lldp)#
```

# protocol spanning-tree

Designates the context for spanning tree.

## Syntax

```
protocol spanning-tree { mstp | rstp | stp | pvst | rpvst }  
no protocol spanning-tree
```

## Command Default

STP is not enabled. STP is not required in a loop-free topology.

## Parameters

<b>mstp</b>	Specifies the Multiple Spanning Tree Protocol (MSTP).
<b>rstp</b>	Specifies the Rapid Spanning Tree (RSTP).
<b>stp</b>	Specifies the Spanning Tree Protocol (STP).
<b>pvst</b>	Specifies Per-VLAN Spanning Tree Protocol Plus (PVST+).
<b>rpvst</b>	Specifies Rapid Per-VLAN Spanning Tree Protocol Plus (R-PVST+).

## Modes

Global configuration mode

## Usage Guidelines

Consider enabling STP to detect or avoid loops. You must turn off one form of STP before turning on another form.

Packet drops or packet flooding may occur if you do not enable xSTP on all devices connected on both sides of parallel links.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

Extreme Network OS supports PVST+ and R-PVST+ only. The PVST and R-PVST protocols are proprietary to Cisco and are not supported.

Enter **no protocol spanning-tree** to delete the context and all the configurations defined within the context or protocol for the interface.

## Examples

To enable the Spanning Tree Protocol:

```
device# configure terminal
device(config)# protocol spanning-tree stp
```

# protocol uddl

Enables and/or enters unidirectional link detection (UDLD) protocol configuration mode.

## Syntax

`protocol uddl`

`no protocol uddl`

## Command Default

This protocol is disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

UDLD detects and blocks a physical link that becomes unidirectional. A unidirectional link can cause traffic in a network to loop endlessly. When the link becomes bidirectional again, UDLD unblocks the link.

This protocol applies only to physical ports. In addition to running this command, you must also enable each desired port for UDLD in interface subconfiguration mode.

Use the `no protocol uddl` command to disable the UDLD protocol and revert all UDLD configuration to defaults.

## Examples

To enable the unidirectional link detection (UDLD) protocol:

```
device# configure terminal
device(config)# protocol uddl
```

## Related Commands

[hello \(UDLD\)](#), [udld enable](#), [shutdown \(UDLD\)](#)



# protocol vrrp

Globally enables VRRP (and VRRP-E on some platforms).

## Syntax

```
protocol vrrp
```

```
no protocol vrrp
```

## Command Default

Disabled

## Modes

RBridge ID configuration mode

## Usage Guidelines

Enables both the VRRP and VRRP-Extended protocols on the Extreme VDX device.

The **no protocol vrrp** command globally disables only VRRP but not VRRP-E.

## Examples

To enable VRRP and VRRP-E:

```
device# configure terminal
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp
```

## Related Commands

[protocol vrrp-extended](#)

# protocol vrrp-extended

Globally enables VRRP-Extended.

## Syntax

```
protocol vrrp-extended  
no protocol vrrp-extended
```

## Command Default

Disabled

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no protocol vrrp-extended** command globally disables VRRP-Extended.

## Examples

To enable VRRP-Extended:

```
device# configure terminal  
device (config)# rbridge-id 101  
device(config-rbridge-id-101)# protocol vrrp-extended
```

## Related Commands

[protocol vrrp](#)

# pwd

Displays the mode of the current working directory.

## Syntax

```
pwd
```

## Modes

Functions in all modes except privileged EXEC mode.

## Examples

To view the current working directory:

```
device# pwd
-----^
syntax error: unknown argument.
device# configure terminal

Entering configuration mode terminal
device(config)# pwd

At top level
device(config)#
```

# python

Launches an interactive Python shell, with an option to launch a Python script.

## Syntax

```
python [ python-statement | python-script-filename ]
```

## Parameters

*python-statement*

Must be a valid python interpreter argument.

*python-script-filename*

Runs a Python script file. Valid values range from 4 through 32 characters (including the **.py** extension). The first character must be alphabetic.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is available only to users with admin-level permissions.

Entering **python**—with no additional parameters—launches an interactive Python shell.

Entering **python** *python-statement* launches an interactive Python shell and runs a valid *python-statement* that you enter. For example, entering `python -h` invokes the Python shell and displays Python options and arguments.

Entering **python** *python-script-filename* launches an interactive Python shell and runs the Python file. (To make a Python file available to this command, copy the Python file to the `flash://` location on the device, using the **copy** command.)

Note the following divergence between Network OS CLI syntax and Python syntax:

- Although in general, Network OS CLI syntax is not case-sensitive, Extreme convention is to use lower-case.
- Python syntax is case sensitive.

To exit the Python environment and return to the CLI, enter either:

- **exit()**
- **Ctrl-D**

## Examples

The following example launches the Python shell and then both assigns a NOS CLI operational command to a Python variable and runs that command.

```
device# python
Python 3.3.2 (default, Apr 11 2014, 13:05:18)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> cmd_show_running_vlan = CLI('show running-config int vlan')
!Command: show running-config int vlan
!Time: Tue Jan 6 00:12:37 2015
interface Vlan 1
!
>>>
```

The following example (partial) launches the Python shell to run a Python script-file.

#### NOTE

For an annotated text of this script, refer to *Network OS Management Configuration Guide* > "Python Event-Management and Scripting" > "Python scripts and run-logs."

```

device# python create_po.py
!Command: show running-config int vlan
!Time: Wed Apr 29 06:54:24 2015

interface Vlan 1
!
interface Vlan 101
!
interface Vlan 102
!
interface Vlan 103
!
interface Vlan 104
!
interface Vlan 105
!

!Command: config
int vlan 101-105
!Time: Wed Apr 29 06:54:25 2015

!Command: show running-config int vlan
!Time: Wed Apr 29 06:54:26 2015

interface Vlan 1
!
interface Vlan 101
!
interface Vlan 102
!
interface Vlan 103
!
interface Vlan 104
!
interface Vlan 105
!

!Command: show running-config int po
!Time: Wed Apr 29 06:54:26 2015

interface Port-channel 10
 vlag ignore-split
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 101-105
 switchport trunk tag native-vlan
 spanning-tree shutdown
 no shutdown
!

!Command: config
int po 10
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 101-105
 switchport trunk tag native-vlan
 no shut
!Time: Wed Apr 29 06:54:27 2015

!Command: show running-config int po
!Time: Wed Apr 29 06:54:27 2015

interface Port-channel 10
 vlag ignore-split

```

```
switchport
switchport mode trunk
switchport trunk allowed vlan add 101-105
switchport trunk tag native-vlan
spanning-tree shutdown
no shutdown
!
```

## History

Release version	Command history
6.0.1	This command was introduced.

## qos

If there is bursty, lossy traffic for certain flows in the system, you can borrow the buffers from less bursty flows, in order to reduce the traffic loss. The **qos** command is used to configure the egress or ingress queue limit (depth), such as the maximum number of kilobytes of data that can be queued in the egress or ingress queue. This configuration is applied on individual RBridges.

## Syntax

```
qos { tx-queue | rcv-queue} [ limit limitInKBytes ]
```

```
no qos
```

## Command Default

The range of queue limit values is from 128 KB through 8 MB. While any value within this range is valid, recommended values are 128, 256, 512, 1024, and 2048.

The default ingress queue limit is 285 kilobytes.

The default egress queue limit is 512 kilobytes.

## Parameters

**tx-queue**

Designates the egress queue.

**rcv-queue**

Designates the ingress queue.

**limit** *limitInKBytes*

Configures the limit of the queue in kilobytes.

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no qos** command removes both queue limits and the default queue limits are restored.

This command only functions on the Extreme VDX 6740 and the Extreme VDX 6740T.

## Examples

This example defines the egress queue to 256 kilobytes.

```
device# configure terminal
device(config)# rbridge-id 154
device(config-rbridge-id-154)# qos tx-queue limit 256
```



## History

Release version	Command history
5.0.0	This command was introduced.
7.0.1	The example was updated.

# qos cos

Changes the interface default Class of Service (CoS) value.

## Syntax

```
qos cos cos_value  
no qos cos
```

## Command Default

The default is 0.

## Parameters

*value*  
Specifies the CoS value. Valid values range from 0 through 7.

## Modes

Interface configuration mode

## Usage Guidelines

When Interface ingress QoS Trust is in the un-trusted mode, then the Interface Default CoS value is applied to all ingress traffic for user priority mapping. When the interface ingress QoS Trust is in the CoS mode, then the Interface Default CoS value is applied to all nonpriority tagged ingress traffic for user priority mapping.

If the interface is QoS trusted, the CoS value of the interface is used to assign a CoS value to all untagged packets entering the interface.

QoS Trust is implicitly turned on when the QoS CoS-Mutation map is applied to interfaces, and is implicitly turned off when the QoS CoS-Mutation map is removed.

Enter **no qos cos** to return the CoS value to the default.

## Examples

To set the CoS value to 2 on a specific 40-gigabit Ethernet interface:

```
device# configure terminal  
device(config)# interface fortygigabitethernet 1/3/1  
device(conf-if-fo-1/3/1)# qos cos 2
```

To return the CoS value to the default on a specific port-channel interface:

```
device# configure terminal  
device(config)# interface port-channel 22  
device(config-port-channel-22)# no qos cos
```

# Related Commands

interface, show qos interface

# qos cos-mutation

Applies a user configured QoS CoS-to-CoS mutation map to an interface.

## Syntax

```
qos cos-mutation cos_map_name
no qos cos-mutation
```

## Command Default

No explicit QoS CoS-to-CoS mutation map is applied; the inbound CoS equals the outbound CoS.

## Parameters

*cos\_map\_name*  
The name of the CoS mutation map.

## Modes

Interface configuration mode

## Usage Guidelines

Mutation mapping is a method of modifying a QoS field in all packets on an interface. On ingress, mutation mapping occurs before traffic classification and all other actions. On egress, mutation mapping occurs after traffic classification and before all other actions.

Enter **no qos cos-mutation** to remove the CoS-to-CoS mutation map.

The **qos cos-mutation** command is not available if the interface is in CEE Provisioning mode.

## Examples

### DSCP-to-DSCP

Follow this example to apply a QoS CoS-to-CoS mutation map to a specific 40-gigabit Ethernet interface

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# qos cos-mutation cos_mutation_map
```

To apply a QoS CoS-to-CoS mutation map to a specific port -channel interface:

```
device# configure terminal
device(config)# interface port-channel 22
device(config-port-channel-22)# qos cos-mutation cos_mutation_map
```

To remove a QoS CoS-to-CoS mutation map from a specific port-channel interface:

```
device# configure terminal
device(config)# interface port-channel 22
device(config-port-channel-22)# no qos cos-mutation
```

## Related Commands

interface, qos map cos-mutation, show qos maps

# qos drop-monitor enable

Under QoS, enables RASlog messages for various types of dropped data.

## Syntax

**qos drop-monitor enable**

**no qos drop-monitor enable**

## Command Default

The QoS drop-monitor feature is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The drop-polling interval is 60 seconds. If drops occur during such an interval, a RASlog message is generated.

The following drop types are logged:

- Extreme VDX 6740 and VDX 6740T: Random Early Detect (RED) and tail drops
- Extreme VDX 8770 (internal port interface ASICs): tail drops only

Use the **no** form of this command to restore the default disablement of this feature.

## Examples

The following example enables RASlog messages for supported types of dropped data.

```
device# configure terminal
device(config)# interface tengigabitethernet 2/2/1
device(conf-if-te-2/2/1)# qos drop-monitor enable
```

## History

Release version	Command history
6.0.0	This command was introduced.

## Related Commands

[interface](#), [qos random-detect traffic-class](#), [qos red profile](#)

# qos dscp-cos

Applies a user configured QoS DSCP-to-CoS mutation map to an interface.

## Syntax

```
qos dscp-cos dscp_cos_map_name
no qos dscp-cos
```

## Command Default

No explicit QoS DSCP-to-CoS mutation map is applied.

## Parameters

*dscp\_cos\_map\_name*  
Name of DSCP-to-COS mutation map

## Modes

Interface configuration mode.

## Usage Guidelines

Mutation mapping is a method of modifying a QoS field in all packets on an interface. On ingress, mutation mapping occurs before traffic classification and all other actions. On egress, mutation mapping occurs after traffic classification and before all other actions.

Enter **no qos dscp-cos** while in the interface mode to remove the DSCP-to-COS mutation map from the interface.

## Examples

Follow this example to apply a user configured QoS DSCP-to-COS mutation map named `dscpMap` to a specific 10-gigabit Ethernet interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 16/2/2
device(conf-if-te-16/2/2)# qos dscp-cos dscpMap
```

Follow this example to apply a user configured QoS DSCP-to-COS mutation map named `dscpMap` to a specific port channel interface.

```
device# configure terminal
device(config)# interface port-channel 22
device(config-port-channel-22)# qos dscp-cos dscpMap
```

Follow this example to remove a user configured QoS DSCP-to-COS mutation map from a specific interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 16/2/2
device(conf-if-te-16/2/2)# no qos dscp-cos
```

# qos dscp-mutation

Applies a user configured QoS DSCP mutation map to an interface.

## Syntax

```
qos dscp-mutation dscp_map_name
no qos dscp-mutation
```

## Command Default

No explicit user configured QoS DSCP-to-DSCP mutation map is applied; the inbound DSCP equals the outbound DSCP.

## Parameters

*dscp\_map\_name*  
The name of the DSCP mutation map

## Modes

Interface subtype configuration mode

## Usage Guidelines

Mutation mapping is a method of modifying a QoS field in all packets on an interface. On ingress, mutation mapping occurs before traffic classification and all other actions. On egress, mutation mapping occurs after traffic classification and before all other actions.

Enter **no qos dscp-mutation** while in the interface mode to remove the DSCP mutation map from the interface.

## Examples

Follow this example to apply a QoS DSCP-to-DSCP mutation map to a specific 40-gigabit Ethernet interface

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/3/1
device(config-if-fo-1/3/1)# qos dscp-mutation dscp_mutation_map
```

To apply a QoS DSCP-to-DSCP mutation map to a specific port channel interface:

```
device# configure terminal
device(config)# interface port-channel 22
device(config-port-channel-22)# qos dscp-mutation dscp_mutation_map
```

To remove a QoS DSCP-to-DSCP mutation map from a specific port channel interface:

```
device# configure terminal
device(config)# interface port-channel 22
device(config-port-channel-22)# no qos dscp-mutation dscp_mutation_map
```



## Related Commands

interface, qos map dscp-mutation, show qos maps dscp-mutation

# qos dscp-traffic-class

Applies a user configured QoS DSCP-to-traffic- class mutation map to an interface.

## Syntax

```
qos dscp-traffic-class dscp_tc_name
no qos dscp-traffic-class
```

## Command Default

No explicit user configured QoS DSCP-to-traffic class map is enabled on the interface.

## Parameters

*dscp\_tc\_name*  
Name of DSCP-to-traffic class map

## Modes

Interface configuration mode

## Usage Guidelines

Mutation mapping is a method of modifying a QoS field in all packets on an interface. On ingress, mutation mapping occurs before traffic classification and all other actions. On egress, mutation mapping occurs after traffic classification and before all other actions.

Enter **no qos dscp-traffic-class** while in the interface mode to remove the DSCP-to-traffic class map from the interface.

## Examples

Follow this example to apply a QoS DSCP-to-traffic class mutation map to a specific 40-gigabit Ethernet interface

```
device# configure terminal
device(config)# interface tengigabitethernet 16/2/2
device(conf-if-te-16/2/2)# qos dscp-traffic-class dscp_tc_map
```

Follow this example to apply a QoS DSCP-to-traffic class mutation map to a specific port channel interface

```
device# configure terminal
device(config)# interface port-channel 22
device(config-port-channel-22)# qos dscp-traffic-class dscp_tc_map
```

Follow this example to remove a configured DSCP-traffic class map from a specific interface.

```
device# configure terminal
device(config)# interface tengigabitethernet 16/2/2
device(conf-if-te-16/2/2)# no qos dscp-traffic-class
```

## Related Commands

interface, qos map dscp-traffic-class, show qos maps dscp-traffic-class

# qos flowcontrol

Activates and configures QoS flow control.

## Syntax

```
qos flowcontrol tx [ on | off ] rx [ on | off ]
```

```
no qos flowcontrol
```

## Parameters

```
tx [ on | off ]
```

Activates or deactivates the transmission portion of flow control.

```
rx [ on | off ]
```

Activates or deactivates the receiving portion of flow control.

## Modes

Interface subtype configuration mode

## Usage Guidelines

When a 1-Gbps local port is already online, and the **qos flowcontrol** command is issued, the pause settings take effect immediately on that local port. However, when the link is toggled, pause is renegotiated. The local port will advertise the most recent **qos flowcontrol** settings. After auto completes, the local port pause settings may change, depending on the outcome of the pause negotiation, per 802.3 Clause 28B, as shown below.

**TABLE 13** Pause negotiation results

Advertised LOCAL cfg	Advertised REMOTE cfg	Negotiated result
Rx=off Tx=on	Rx=on Tx=on	asymmetrical: LOCAL Tx=on --> pause --> REMOTE Rx=on
Rx=on Tx=on	Rx=off Tx=on	asymmetrical: LOCAL Rx=on <-- pause <-- REMOTE Tx=on
Rx=on Tx=n/a	Rx=on Tx=n/a	symmetrical : LOCAL Tx/Rx=on <-- pause -- > REMOTE Tx/Rx=on
Rx=n/a Tx=n/a	Rx=off Tx=off	disable pause both sides

Enter **no qos flowcontrol** to deactivate flow control on a specific interface.

## Examples

To activate both the transmitting and receiving portions of flow control on a specific 40-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/3/1
device(conf-fo-1/3/1)# qos flowcontrol tx on rx on
```

To deactivate flow control on a specific port-channel interface:

```
device# configure terminal
device(config)# interface port-channel 33
device(config-port-channel-33)# no qos flowcontrol
```

## Related Commands

[interface](#)

# qos map cos-mutation

Creates a QoS map for performing CoS-to-CoS mutation.

## Syntax

```
qos map cos-mutation name cos0 cos1 cos2 cos3 cos4 cos5 cos6 cos7
no qos map cos-mutation name
```

## Command Default

No CoS-to-CoS mutation QoS maps are defined.

## Parameters

*name*

Specifies a unique name across all CoS-to-CoS mutation QoS maps defined within the system. If the named CoS-to-CoS mutation QoS map does not exist, then it is created. If the named CoS-to-CoS mutation QoS map already exists, then it is updated and new mapping is automatically propagated to all interfaces bound to the QoS map.

*cos#*

Specifies the outbound CoS value.

CoS value	Description
<i>cos0</i>	Sets the outbound CoS value for all packets with inbound CoS 0.
<i>cos1</i>	Sets the outbound CoS value for all packets with inbound CoS 1.
<i>cos2</i>	Sets the outbound CoS value for all packets with inbound CoS 2.
<i>cos3</i>	Sets the outbound CoS value for all packets with inbound CoS 3.
<i>cos4</i>	Sets the outbound CoS value for all packets with inbound CoS 4.
<i>cos5</i>	Sets the outbound CoS value for all packets with inbound CoS 5.
<i>cos6</i>	Sets the outbound CoS value for all packets with inbound CoS 6.
<i>cos7</i>	Sets the outbound CoS value for all packets with inbound CoS 7.

## Modes

Global configuration mode

## Usage Guidelines

A CoS-to-CoS mutation takes an inbound CoS value and maps it to an outbound CoS value. The inbound CoS value is the user priority after any interface ingress QoS trust and Interface default CoS policy have been applied. The outbound CoS value is used in selecting Traffic Class and egress packet marking.

Enter **no qos map cos-mutation *name*** command to delete the named CoS-to-CoS mutation QoS map. A QoS map can only be deleted if it is not bound to any interface.

## Examples

To create a CoS-to-CoS QoS mutation map to swap CoS 4 and CoS 5 and apply it on an interface, for example having inbound CoS 4 mapped to outbound CoS 5 and inbound CoS 5 mapped to outbound CoS 4; but all other CoS values go through unchanged:

```
device# configure terminal
device(config)# qos map cos-mutation 0 1 2 3 5 4 6 7
device(config)# interface tengigabitethernet 1/0/1
device(conf-if-te-1/0/1)# qos cos-mutation cosMap
```

To delete a CoS-to-CoS QoS mutation map:

```
device# configure terminal
device(config)# no qos map cos-mutation cosMap
```

## Related Commands

[qos cos-mutation](#), [show qos maps](#)

# qos map dscp-cos

Creates a QoS map where the ingress DSCP value is mapped to outgoing 802.1P values. This configures a DSCP-to-CoS map on the ingress interface.

## Syntax

```
qos map dscp-cos name  
no qos map dscp-cos name  
mark ingress dscp values to cos
```

## Command Default

DSCP-to-CoS mutation is not enabled.

## Parameters

*name*  
Name of DSCP-to-CoS map

*ingress dscp values*  
Input DSCP values. The range of ingress DSCP values is 0 through 63.

*cos*  
CoS value. The range is 0 through 7.

## Modes

dscp-cos mode for the QoS **mark** commands  
Global configuration mode

## Usage Guidelines

This command remaps the incoming DSCP values of the ingress packet to egress CoS 802.1P values.

When you enter **qos map dscp-cos**, the system is placed in dscp-cos mode for the configured map. At this point, you can map ingress DSCP values to egress CoS values using the **mark** command.

Enter **qos dscp-cos name** while in configuration mode for a specific interface to apply the DSCP-to-CoS map to that interface.

Enter **no qos dscp-cos name** while in the interface configuration mode to remove the DSCP-to-CoS map from the interface.

Enter **no map dscp-cos name** while in global configuration mode to remove the DSCP-to-CoS map.



## Examples

To create a QoS DSCP-to-CoS map and place system into dscp-cos mode:

```
device# configure terminal
device(config)# qos map dscp-cos test
device(dscp-cos-test)#
```

To map an ingress DSCP value to egress CoS values while in dscp-cos mode:

```
device# configure terminal
device(config)# qos map dscp-cos test
device(dscp-cos-test)# mark 1,3,5,7 to 3
```

To map multiple ingress DSCP values to egress CoS values while in dscp-cos mode:

```
device# configure terminal
device(config)# qos map dscp-cos test
device(dscp-mutation-test)# mark 1,3,5,7 to 9
device(dscp-mutation-test)# mark 11,13,15,17 to 5
device(dscp-mutation-test)# mark 12,14,16,18 to 6
device(dscp-mutation-test)# mark 2,4,6,8 to 7
```

To remove a QoS DSCP-CoS map while in global configuration mode:

```
device# configure terminal
device(config)# no qos map dscp-cos test
```

## Related Commands

[qos map dscp-traffic-class](#), [qos dscp-cos](#), [show qos maps dscp-cos](#)

# qos map dscp-mutation

Creates a DSCP mutation by mapping the incoming DSCP value of the ingress packet to outgoing DSCP values.

## Syntax

```
qos map dscp-mutation name
no map qos dscp-mutation name
mark ingress dscp values to egress dscp value
```

## Command Default

DSCP mutation is not enabled.

## Parameters

*name*  
Name of DSCP mutation map

*ingress dscp values*  
The ingress DSCP values. The range is from 0 through 63.

*egress dscp values*  
The egress DSCP value. The range is from 0 through 63.

## Modes

dscp-mutation mode for the DSCP mutation map  
Global configuration mode

## Usage Guidelines

Enter **qos dscp-mutation** *name* while in configuration mode for a specific interface to apply the DSCP mutation map to that interface. When you enter **qos map dscp-mutation**, the system is placed in dscp-mutation mode for the configured map. At this point, you can map ingress DSCP values to egress DSCP values using the **mark** command.

Enter **no qos dscp-mutation** *name* while in interface configuration mode to remove the DSCP mutation map from that interface.

Enter **no map dscp-mutation** *name* while in global configuration mode to remove the DSCP mutation map.

### NOTE

This command is only supported on VDX 8770-4, VDX 8770-8, and later devices.

## Examples

To create a QoS DSCP mutation map and place system into dscp-mutation mode:

```
device# configure terminal
device(config)# qos map dscp-mutation test
device(dscp-mutation-test)#
```

To map an ingress DSCP value to egress DSCP values while in dscp-mutation mode:

```
device# configure terminal
device(config)# qos map dscp-mutation test
device(dscp-mutation-test)# mark 1,3,5,7 to 9
```

To map multiple ingress DSCP values to egress DSCP values while in dscp-mutation mode:

```
device# configure terminal
device(config)# qos map dscp-mutation test
device(dscp-mutation-test)# mark 1,3,5,7 to 9
device(dscp-mutation-test)# mark 11,13,15,17 to 19
device(dscp-mutation-test)# mark 12,14,16,18 to 20
device(dscp-mutation-test)# mark 2,4,6,8 to 10
```

To remove a QoS DSCP mutation map while in global configuration mode:

```
device# configure terminal
device(config)# no qos map dscp-mutation test
```

## Related Commands

[qos dscp-mutation](#), [show qos maps dscp-mutation](#)

# qos map dscp-traffic-class

Creates a QoS map for performing DSCP-to-traffic class mapping. This creates a DSCP-to-traffic class map on the ingress interface. You can configure an interface with either a DSCP-to-traffic class map or a CoS-to-traffic class map.

## Syntax

```
qos map dscp-traffic-class name
no qos map dscp-traffic-class name
mark ingress dscp values to traffic class
```

## Command Default

DSCP-to-traffic class mutation is not enabled.

## Parameters

*name*  
Name of the QoS DSCP-to-traffic clas map.

**traffic-class**  
Egress traffic class values. The range of ingress traffic class values is from 0 through 7.

*ingress dscp values*  
Range of input DSCP values. The range is 0 through 63.

*traffic class*  
The traffic class value. the range is from 0 through 7.

## Modes

dscp-traffic-class mode for the DSCP-to-traffic class map  
Global configuration mode

## Usage Guidelines

Enter **qos dscp-traffic-class** *name* while in configuration mode for a specific interface to apply the QoS DSCP-Traffic-Class map to that interface. When you enter **qos map dscp-traffic-class**, the system is placed in dscp-traffic-class mode for the configured map. At this point, you can map ingress DSCP values to traffic class values using the **mark** command.

Enter **no qos dscp-traffic-class** *name* while in the interface mode to remove the map from that interface.

Enter **no map dscp-mutation** *name* to remove the map while in global configuration mode.

## Examples

To create a QoS DSCP-to-traffic class map and place system into dscp-traffic-class mode:

```
device# configure terminal
device(config)# qos map dscp-traffic-class test
device(dscp-traffic-class-test)#
```

To map ingress DSCP values to a traffic class while in dscp-traffic-class mode:

```
device# configure terminal
device(config)# qos map dscp-traffic-class test
device(dscp-traffic-class-test)# mark 1,3,5,7 to 3
```

To map multiple ingress DSCP values to traffic classes while in dscp-traffic-class mode:

```
device# configure terminal
device(config)# qos map dscp-traffic-class test
device(dscp-traffic-class-test)# mark 1,3,5,7 to 3
device(dscp-traffic-class-test)# mark 11,13,15,17 to 5
device(dscp-traffic-class-test)# mark 12,14,16,18 to 6
device(dscp-traffic-class-test)# mark 2,4,6,8 to 7
```

To remove a QoS DSCP-traffic class map while in global configuration mode:

```
device# configure terminal
device(config)# no qos map dscp-traffic-class test
```

## Related Commands

[qos dscp-traffic-class](#), [show qos maps dscp-traffic-class](#)

# qos random-detect traffic-class

Maps a Random Early Detect (RED) profile to a CoS priority value for a port.

## Syntax

```
qos random-detect traffic-class value red-profile-id profile-ID value  
no qos random-detect traffic-class value
```

## Command Default

Port CoS priority value is not mapped to the RED profile.

## Parameters

*value*

Class of Service (COS) value. Valid values range from 0 through 7.

*profile-ID* *value*

Random Error Detection value. Valid values range from 1 through 384.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The RED profile is defined by the **qos red-profile** command.

Enter **no qos random-detect traffic-class *value*** while in the interface mode to remove the DSCP-to-Traffic-Class map from the interface.

## Examples

To map the profile to CoS priority 7 on the 10-gigabit Ethernet interface 1/2/2:

```
device# configure terminal  
device(config)# interface tengigabitethernet 1/2/2  
device(conf-if-te-1/2/2)# qos random-detect traffic-class 7 red-profile-id 2
```

To remove the previously created profile from interface 1/2/2:

```
device# configure terminal  
device(config)# interface tengigabitethernet 1/2/2  
device(conf-if-te-1/2/2)# no qos random-detect traffic-class 7
```

# qos rcv-queue limit

Controls high burst traffic received on the Extreme VDX 6740 and VDX 6940.

## Syntax

```
qos rcv-queue limit { buffering_upper_limit }
```

## Command Default

The default value is 2048 KB.

## Parameters

*buffering\_upper\_limit*

Defines the upper limit of buffering for the port. The range of queue limit values is from 128 KB through 8 MB. While any value within this range is valid, recommended values are 128, 256, 512, 1024, and 2048.

## Modes

RBridge ID configuration mode

## Usage Guidelines

With enhanced shared dynamic buffering mechanism, an interface is capable of bursting up to the recommended 2MB limit. Though a maximum of 8MB is allowed, you should consult your Extreme Engineer, as it may impact the performance of the other ports that may need to burst at the same time.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-1)# qos rcv-queue limit 8000
```

## History

Release version	Command history
4.0.1	This command was introduced.

# qos red profile

Creates a Random Early Detect (RED) profile for egress traffic flow and provides a minimum threshold, maximum threshold, and drop-probability for egress traffic flow.

## Syntax

```
qos red-profile profile-ID value min-threshold percentage max-threshold percentage drop-probability percentage
no qos red-profile profile-IDvalue
```

## Parameters

*profile-ID value*

Valid values range from 1 through 384.

*percentage*

0 through 100 percent.

**min-threshold**

Minimum threshold (percentage) of queue size (0 through 100) for randomly dropping packets.

**max-threshold**

Maximum threshold (percentage) of queue size (0 through 100) when packets are dropped with 100% probability.

**drop-probability**

Probability that packets will be dropped when minimum threshold is reached.

## Modes

Global configuration mode

## Usage Guidelines

Enter **qos random-detect cos** command while in configuration mode for a specific interface to map the profile to a CoS priority for a port.

Enter **no qos random-detect cos** command while in the interface mode to remove the profile from the interface. You must remove the profile from interface before you can remove the profile itself.

Enter **no qos red-profile *profile-ID*** to remove the profile while in global configuration mode.

### NOTE

This command is only supported on Extreme VDX 6740, VDX 6790, VDX 8770, and later devices.

## Examples

To create a RED profile while in global configuration mode:

```
device# configure terminal
Entering configuration mode terminal
device(config)# qos red-profile 2 min-threshold 10 max-threshold 80 drop-probability 80
```



To remove the profile while in global configuration mode:

```
device# configure terminal
device(config)# no qos red-profile 2
```

## Related Commands

[qos drop-monitor enable](#), [show qos red profiles](#)

# qos service-policy

Activates the global service policy mode that allows you to apply a service policy to a single Rbridge ID, a range of Rbridge IDs, or all Rbridge IDs.

## Syntax

```
qos service-policy { in | out } service_policy_name
no qos service-policy { in | out } service_policy_name
```

## Parameters

**in**  
Applies the service policy to inbound traffic.

**out**  
Applies the service policy to outbound traffic.

*service\_policy\_name*  
The name of the service policy to apply to the Rbridge ID.

## Modes

Global configuration mode.

## Usage Guidelines

The policy map has been preconfigured.

Enter **no qos service-policy in/out service\_policy\_name** to return to the default.

## Examples

This example applies a service policy to inbound traffic on all Rbridge IDs.

```
device# configure terminal
device(config)# qos service-policy in policymap1
device(config-service-policy)# attach rbridge-id add all
```

This example removes a service policy on all Rbridge IDs.

```
device# configure terminal
device(config)# qos service-policy in policymap1
device(config-service-policy)# attach rbridge-id remove all
```

## History

Release version	Command history
5.0.0	This command was introduced.

# qos tx-queue limit

Controls high burst traffic transmitted on the Extreme VDX 6740 and VDX 6940.

## Syntax

```
qos tx-queue limit { buffering_upper_limit }
```

## Command Default

The default value is 1024 KB.

## Parameters

*buffering\_upper\_limit*

Defines the upper limit of buffering for the port.

The range of queue limit values is from 128 KB through 8 MB. While any value within this range is valid, recommended values are 128, 256, 512, 1024, and 2048.

## Modes

RBridge ID configuration mode

## Usage Guidelines

With enhanced shared dynamic buffering mechanism, an interface is capable of bursting up to the recommended 2MB limit. Though a maximum of 8MB is allowed, you should consult your Extreme Engineer, as it may impact the performance of the other ports that may need to burst at the same time.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-1)# qos tx-queue limit 8000
```

## History

Release version	Command history
4.0.1	This command was introduced.

# qos-profile (AMPP)

Activates the QoS profile mode for AMPP. This mode allows configuration of QoS attributes of a port-profile.

## Syntax

**qos-profile**

**no qos-profile**

## Modes

Port-profile configuration mode

## Usage Guidelines

Enter **no qos-profile** to remove the profile.

## Examples

```
device# configure terminal
device(config)# port-profile sample-profile
device(conf-pp)# qos-profile
```

# radius-server

Configures the Remote Authentication Dial-In User Service (RADIUS) server.

## Syntax

```
radius-server host { ip-address | host_name } [ auth-port portnum ] [ protocol { chap | pap | peap } ] [ key shared_secret ]
  [ encryption-level value_level ] [ timeout sec ] [ retries num ] [ use-vrf vrf-name ]
no radius-server host hostname | ip-address [ use-vrf vrf-name ]
```

## Command Default

A Remote Authentication Dial-In User Service (RADIUS) server is not configured.

## Parameters

**host** { *ipaddr* | *host\_name* }

Specifies the IP address or host name of the RADIUS server. IPv4 and IPv6 addresses are supported. The maximum supported length for the RADIUS hostname is 40 characters.

**auth-port** *portnum*

Specifies the user datagram protocol (UDP) port used to connect the RADIUS server for authentication. The valid range is 0 through 65535. The default port is 1812.

**protocol** { *chap* | *pap* | *peap* }

Specifies the authentication protocol. Parameters include CHAP, PAP, or PEAP-MSCHAP. The default is CHAP.

**key** *shared\_secret*

The text string that is used as the shared secret between the device and the RADIUS server. The default is **sharedsecret**. The exclamation mark (!) is supported both in RADIUS and TACACS+ servers, and you can specify the shared secret string in either double quotation marks or use the escape character (\). For example: **"secret!key"** or **secret\!key**.

**encryption-level** *value\_level*

Designates the encryption level for the shared secret key operation. This operand supports J1TC certification and compliance. The valid values are 0 and 7, with 0 being clear text and 7 being the most heavily encrypted. The default value is 7.

**timeout** *sec*

The time to wait for the RADIUS server to respond, in seconds. The default is 5 seconds.

**retries** *num*

The number of attempts allowed to connect to a RADIUS server. The default is 5 attempts.

**use-vrf** *vrf-name*

Specifies a VRF through which to communicate with the RADIUS server. See the Usage Guidelines.

## Modes

Global configuration mode

## Usage Guidelines

If a RADIUS server with the specified IP address or host name does not exist, it is added to the server list. If the RADIUS server already exists, this command modifies the configuration.

The **key** parameter does not support an empty string.

Enter **no radius-server** to reset to their default values.

### NOTE

Before downgrading to a software version that does not support the **encryption-level** keyword, set the value of this keyword to 0. Otherwise, the firmware download will throw an error that requests this value be set to 0.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

To configure a RADIUS server:

```
device# configure terminal
device(config)# radius-server host 10.24.65.6 protocol chap retransmit 100
device(config-radius-server-10.24.65.6)#
```

To modify the previously configured RADIUS server:

```
device# configure terminal
device(config)# radius-server host 10.24.65.6 protocol pap key "new#radius*secret" timeout 10
```

To reset the timeout value to the default:

```
device# configure terminal
device(config)# no radius-server host 10.24.65.6 timeout
```

To communicate with the server through a user-specified VRF:

```
device# configure terminal
device(config)# radius-server host 10.24.65.6 use-vrf my-vrf
device(config-host-10.24.65.6/mgmt-vrf)#
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.

## Related Commands

[show running-config radius-server](#), [show running-config tacacs-server](#), [tacacs-server](#)

# rasman

Displays RASLog messages decoding and documentation on the switch.

## Syntax

```
rasman [[ module-description ]][ [ message id RAS-message-id ]][ [ module type module-name ]][ [ type value RAS-message-type ]]
```

## Parameters

### module-description

Displays the RAS module description.

### message id *RAS-message-id*

Displays the RAS message ID details.

### module type *module-name*

Displays the RAS message ID based on module. Displays all external RAS messages.

### type value *RAS-message-type*

Displays the RAS message ID based on type. Possible completions: AUDIT, DCE, FFDC, LOG, and VCS.

## Modes

Privileged EXEC mode

## Usage Guidelines

Input value is case-sensitive.

## Examples

To display the module descriptions:

```
device# rasman module-description
RASModule      ID      Description
-----
KT              1      Kernel Test ID
UT              2      User Test ID
TRCE           3      Trace Subsystem (User)
KTRC           4      Trace Subsystem (Kernel)
LOG            5      RASLOG module
CDR            6      Condor ASIC driver
```

To display the messages pertaining to the AUTH module:

```
device# rasman module type AUTH
RAS Message ID      Severity      Message
-----
AUTH-1001           INFO          %s has been successfully completed.
AUTH-1002           ERROR         %s has failed.
AUTH-1003           INFO          %s type has been successfully set t
AUTH-1004           ERROR         Failed to set %s type to %s.
AUTH-1005           ERROR         Authentication file does not exist:
AUTH-1006           WARNING      Failed to open authentication confi
AUTH-1007           ERROR         The proposed authentication protoco
AUTH-1008           ERROR         No security license, operation fail
```

To display the AUDIT messages.

```
device# rasman type value AUDIT
RAS Message ID      Severity      Message
-----
FCIP-1002           INFO          An IPsec/IKE policy was added
FCIP-1003           INFO          An IPsec/IKE policy was deleted
AUTH-1045           ERROR         Certificate not present in this switch
AUTH-1046           INFO          %s has been successfully completed
AUTH-1047           ERROR         %s has failed
AUTH-3001           INFO          Event: %s, Status: success, Info: %
AUTH-3002           INFO          Event: %s, Status: success, Info: %
```

Example of rasman type value DCE.

```
device# rasman type value DCE
RAS Message ID      Severity      Message
-----
LACP-1001           ERROR         %s Error opening socket (%d)
LACP-1002           ERROR         %s %s
LACP-1003           INFO          Port-channel %d up in defaulted state
LACP-1004           INFO          Port-channel %d down from default
NSM-1001           INFO          Interface %s is online
NSM-1002           INFO          Interface %s is protocol down
```

Example of rasman type value LOG:

```
device# rasman type value LOG
RAS Message ID      Severity      Message
-----
FCIP-1000           ERROR         %s of GE %d failed. Please retry
FCIP-1001           CRITICAL     FIPS %s failed; algo=%d type=%d slot
FCIP-1002           INFO          An IPsec/IKE policy was added
FCIP-1003           INFO          An IPsec/IKE policy was deleted
FCIP-1004           INFO          Tape Read Pipelining is being disabled
AUTH-1001           INFO          %s has been successfully completed
AUTH-1002           ERROR         %s has failed
```

Example of rasman type value VCS:

```
device# rasman type value VCS
RAS Message ID      Severity      Message
-----
SS-2000           INFO          Copy support started on rbridge-id
SS-2001           INFO          Copy support completed on rbridge-id
SS-2002           INFO          Copy support failed on rbridge-id %
SULB-1105           WARNING      Firmware upgrade session (%d: %s) s
SULB-1106           WARNING      Firmware upgrade session (%d: %s) c
SULB-1107           WARNING      Firmware upgrade session (%d: %s) f
```



# rate-limit-delay get netconf

Returns the rate limit delay configured for processing NETCONF Remote Procedure Calls (RPCs).

## Syntax

```
rate-limit-delay get netconf
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command returns the configured minimum time in milliseconds between processing successive NETCONF RPCs. A value of 0 indicates that RPC processing is unlimited.

## Examples

This example limits the processing of RPCs to a maximum of one every 50 milliseconds.

```
device# rate-limit-delay get netconf
```

## Related Commands

[rate-limit-delay set netconf](#)

# rate-limit-delay set netconf

Limits the rate at which BNA or NETCONF Remote Procedure Call (RPC) requests can be processed on the device.  
Synopsis: `rate-limit-delay set netconf delay`

## Syntax

`rate-limit-delay set netconf value`

## Command Default

The default is 0.

## Parameters

*value*

The number of milliseconds the system waits between processing RPCs.

## Modes

Privileged EXEC mode

## Usage Guidelines

The rate at which RPCs can be processed on the device is specified the minimum delay between processing successive RPCs. The default of 0 means that the RPC processing rate is unlimited.

Use this command to prevent excessive numbers of RPCs from adversely affecting device performance.

## Examples

This example limits the processing of RPCs to a maximum of one every 50 milliseconds.

```
device# debug internal rate-limit-delay set netconf 50
```

## Related Commands

[rate-limit-delay get netconf](#)

# rbridge-id

Enables RBridge ID configuration mode to support VCS on individual nodes.

## Syntax

```
rbridge-id rbridge-id
```

```
no rbridge-id rbridge-id
```

## Parameters

*rbridge-id*

The number of an RBridge node.

## Modes

Global configuration mode

## Usage Guidelines

### ATTENTION

It is always preferable to have the RBridge ID configured on a switch. If the RBridge ID is not configured, deleting the interface IP address that matches the router ID will cause an OSPF process reset and a spike in CPU usage.

## Examples

Use the `rbridge-id ?` command in global configuration mode to see available nodes.

Enter RBridge ID configuration mode and use `?` to view commands available in that mode:

```
device# configure terminal
device(config)# rbridge-id 154
device(config-rbridge-id-154)#
```

## rd (VNI)

Configures a route distinguisher (RD) for a virtual network identifier (VNI) under an Ethernet Virtual Private Network (EVPN) instance.

### Syntax

```
rd admin-value:arbitrary-value  
no rd admin-value:arbitrary-value
```

### Command Default

Disabled.

### Parameters

*admin-value*

The administrative number assigned to the route. This can be a local ASN number or an IP address. The ASN number can be either a 2-byte number (from 0 through 65535) or a 4-byte number (from 0 through 4294967295).

*arbitrary-value*

An arbitrary number you choose. The range of valid values is from 0 through 65535 if the ASN is 2 byte, or from 0 through 4294967295 if the ASN is 4 byte.

### Modes

VNI configuration mode

### Usage Guidelines

The Route Distinguisher parameter can be either ASN-relative or IP address-relative.

The **no** form of the command in VNI configuration mode removes the configuration for a particular VNI under an EVPN instance.

### Examples

The following example configures an RD and assigns the IP address 10.2.1.1:1 for VNI 100.

```
device# configure terminal  
device(config)# rbridge-id 1  
device(config-rbridge-id-1)# evpn-instance myinstance  
device(config-evpn-instance-myinstance)# vni 100  
device(evpn-vni-100)# rd 10.2.1.1:1
```

# History

Release version	Command history
7.0.0	Support was added for VNI configuration mode.

# rd (VRF)

Configures a route distinguisher (RD) for a VRF instance.

## Syntax

```
rd admin-value:arbitrary-value
```

## Parameters

*admin-value*

The administrative number assigned to the route. This can be a local ASN number or an IP address. The ASN number can be either a 2-byte number (from 0 through 65535) or a 4-byte number (from 0 through 4294967295).

*arbitrary-value*

An arbitrary number you choose. The range of valid values is from 0 through 65535 if the ASN is an IP address or a 2 byte ASN. The range is 0 through 4294967295 if the ASN is a 4 byte ASN.

## Modes

VRF configuration mode

## Usage Guidelines

This command allows you to use the same IP address in different VPNs without creating any conflicts. The Route Distinguisher parameter can be either ASN-relative or IP address-relative.

Once the Route Distinguisher is configured for a VRF it cannot be changed or deleted. To remove the Route Distinguisher, you must delete the VRF.

## Examples

The following example configures an RD and assigns the local ASN number 101:1 for VRF red.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf red
device(config-vrf-red)# rd 101:1
```

The following example configures an RD and assigns the IP address 10.1.1.1:1 for VRF red.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf red
device(config-vrf-red)# rd 10.1.1.1:1
```

The following example removes an RD for VRF red.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# no vrf red
```

## Related Commands

ip router-id, vni (EVPN), vrf

# rd auto (EVPN)

Enables auto-generation of a route distinguisher (RD) for an Ethernet Virtual Private Network (EVPN) instance.

## Syntax

```
rd auto
no rd auto
```

## Command Default

Disabled.

## Modes

EVPN instance configuration mode

## Usage Guidelines

This command allows you to automatically create RDs. An RD is created for each VNI that is added for that EVPN Instance using the **vni** command.

The **no** form of the command disables auto-generation of an RD for an EVPN instance.

## Examples

The following example enables auto-generation of the RD value for EVPN instance "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# rd auto
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[evpn-instance](#)



# reconnect-interval

Sets the reconnect interval between the NSX controller and the VCS fabric.

## Syntax

```
reconnect-interval interval
```

```
no reconnect-interval
```

## Command Default

10 seconds

## Parameters

*interval*

Specifies the maximum number of seconds to wait between connection attempts. Value must be in the range of 1 to 1000.

## Modes

NSX controller configuration mode

## Usage Guidelines

Use this command to set the reconnect interval for the NSX controller connection profile. If the connection is lost between the NSX and the VCS fabric, a reconnection attempt occurs at this interval.

The **no** form of the command reverts the reconnect interval to the default value.

## Examples

The following example sets the reconnect interval to 30 seconds for an NSX controller profile that you have already created (named *profile1*).

```
device# configure terminal
device(config)# nsx-controller profile1
device(config-nsx-controller-profile1)# reconnect-interval 30
```

# redistribute

Configures the device to redistribute IPv4 and IPv6 routes from one routing domain to another.

## Syntax

```
redistribute ospf [ match { external1 | external2 | internal } | metric num | metric-type { type1 | type2 } | route-map string ]
redistribute { source-protocol } [ metric num | metric-type { type1 | type2 } | route-map string ]
no redistribute ospf [ match { external1 | external2 | internal } | metric num | metric-type { type1 | type2 } | route-map string ]
no redistribute { source-protocol } [ metric num | metric-type { type1 | type2 } | route-map string ]
```

## Command Default

The device does not redistribute routing information.

## Parameters

### ospf

Specifies the OSPF protocol.

### match

Specifies the type of route.

#### external1

Specifies OSPF Type 1 external routes.

#### external2

Specifies OSPF Type 2 external routes.

#### internal

Specifies OSPF internal routes.

### metric *num*

Specifies a metric for redistributed routes. Range is from 0 through 65535. No value is assigned by default.

### metric-type

Specifies the external link type associated with the default route advertised into the OSPF routing domain.

#### type1

Specifies a type 1 external route.

#### type2

Specifies a type 2 external route.

### route-map *string*

Specifies a route map to be consulted before a route is added to the routing table.

### *source-protocol*

Specifies the source protocol from which routes are being redistributed. It can be one of the following keywords: **bgp**, **connected**, or **static**.

## Modes

BGP address-family IPv4 unicast configuration mode  
 BGP address-family IPv6 unicast configuration mode  
 BGP address-family IPv4 unicast VRF configuration mode  
 BGP address-family IPv6 unicast VRF configuration mode  
 OSPF router configuration mode  
 OSPFv3 router configuration mode  
 OSPF router VRF configuration mode  
 OSPFv3 router VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

Routes can be filtered by means of an associated route map before they are distributed.

The **metric-type { type1 | type2 }** option is only available in OSPFv3 VRF configuration mode.

[ match metricmetric-type

### NOTE

The **default-metric** command does not apply to the redistribution of directly connected routes. Use a route map to change the default metric for directly connected routes.

## Examples

The following example redistributes OSPF external type 1 routes with a metric of 200.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# redistribute ospf match external1 metric 200
```

The following example redistributes OSPFv3 external type 2 routes in VRF instance "red":

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# redistribute ospf match external2
```

The following example redistributes static routes into BGP4 and specifies a metric of 200.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# redistribute static metric 200
```

The following example redistributes directly connected routes into BGP4+ in VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# redistribute connected
```

The following example redistributes BGP4 routes and specifies that route-map "rm2" be consulted in BGP address-family IPv4 unicast configuration mode.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# redistribute bgp metric 10
```

The following example redistributes BGP routes and specifies that route-map "rm7" be consulted in OSPF VRF configuration mode.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router ospf
device(config-router-ospf-vrf-default-vrf)# redistribute bgp route-map rm7
```

The following example redistributes OSPF routes and specifies a type1 external route in OSPFv3 VRF configuration mode.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# redistribute ospf metric-type type1
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes. Support was added OSPF VRF and OSPFv3 configuration modes.

## Related Commands

[route-map](#), [default-metric \(OSPF\)](#)

# region

Assigns a name to a Multiple Spanning Tree Protocol (MSTP) region.

## Syntax

**region** *region-name*

**no region**

## Parameters

*region-name*

Assigns a name to an MSTP region.

## Modes

Spanning tree MSTP configuration mode

## Usage Guidelines

The *region-name* string must be between 1 and 32 ASCII characters in length, and is case-sensitive.

Enter **no region** to delete the region name.

If MSTP is enabled over VCS, this command must be executed on all RBridge nodes

## Examples

To assign a name to an MSTP region named extreme1:

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# region extreme1
```

## Related Commands

[revision](#), [show spanning-tree](#)

# relay

Configures the IP relay address for Monitoring and Alerting Policy Suite (MAPS) notifications.

## Syntax

```
relay {ip_address } [domainname| domain-name]
no relay {ip_address } [domainname| domain-name]
```

## Parameters

**relay** *ip\_address*  
Destination relay for MAPS notifications.

**domainname** *domain-name*  
Destination domain name for MAPS notifications.

## Modes

MAPS configuration mode

## Usage Guidelines

Use the **no relay** command to remove the relay IP address.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# maps
device(config-rbridge-id-5-maps)# relay 10.25.248.25 domainname abc123.com
```

## History

Release version	Command history
6.0.1	This command was introduced.

# reload

Reboots the control processor (CP) or management module (MM).

## Syntax

```
reload [ standby | system ]
```

```
reload system [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### standby

Reboots the standby CP or MM on a dual CP/MM chassis.

### system

Reboots an entire chassis.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command performs a "cold reboot" (power off and restart) of the CP or MM.

The **reload** operation is generally disruptive and the command prompts for confirmation before executing. When you reboot a device all traffic to and from that device stops. All ports on that device remain inactive until the device comes back online.

On a Top-of-Rack (ToR) device, If the power-on-self-test (POST) is enabled, it is executed when the system comes back up.

On a modular chassis, the **reload** command only reboots the management module on which the command is executed. If you log in to the device IP address and execute the **reload** command, only the active management module reboots and POST is bypassed.

The following summarizes the behavior of the **reload** command under a variety of conditions.

**TABLE 14** Behavior of the **reload** command

Command	HA synchronized		HA not synchronized	
	Active	Standby	Active	Standby
reload	If executed on active MM, reboots that MM 1.	If executed on active MM, reboots that MM.	The user is prompted to execute the <b>reload system</b> command.	N/A

**TABLE 14** Behavior of the **reload** command (continued)

Command	HA synchronized		HA not synchronized	
		The running configuration becomes the new active configuration.		
reload standby	Reboots the standby MM.	Reboots the standby MM.	Reboots the standby MM.	Reboots the standby MM.
reload system	Reboots the chassis and remains the master MM.	Not allowed.	If executed on active MM, reboots the chassis and remains the master MM.	Not allowed.
	The running configuration is used.		The running configuration is used.	

## Examples

The following example performs a cold reboot on the device.

```
device# reload
Are you sure you want to reload the switch [y/n]?: y
```

## History

Release version	Command history
6.0.1a	This command was modified to remove reference to "standalone" mode.

## Related Commands

[fastboot](#), [ha chassisreboot](#), [ha failover](#)



# remap fabric-priority

Remaps the CoS fabric priority to a different priority for Extreme VCS Fabric mode.

## Syntax

```
remap fabric-priority priority
```

## Command Default

The default is 0.

## Parameters

*priority*

Specifies the remapped CoS priority value for Extreme VCS Fabric mode. The valid range is 0 through 6.

## Modes

CEE map configuration mode

## Examples

To set the configuration revision to 1:

```
device# configure terminal
device(config)# cee-map default
device(config-cee-map-default)# remap lossless-priority priority 2
device(fabric-cee-map-default)# remap fabric-priority priority 2
```

# remap lossless-priority

Remaps the Extreme VCS Fabric lossless priorities to a different priority

## Syntax

```
remap lossless-priority priority
```

## Parameters

*priority*

Specifies the remapped priority value. Valid values range from 0 through 6. Default is 0.

## Modes

CEE map configuration mode

## Examples

To set the configuration revision to 1:

```
device# configure terminal
device(config)# cee-map default
device(config-cee-map-default)# remap lossless-priority priority 2
```

# rename

Renames a file in the device flash memory.

## Syntax

```
rename current_name new_name
```

## Parameters

*current\_name*

Specifies the file name you want to change.

*new\_name*

Specifies the new file name.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local device.

System configuration files cannot be renamed. If you try to rename a system file, a warning message is displayed.

## Examples

To rename a file:

```
device# rename myconfig.vcs myconfig.old
device# dir

total 24
drwxr-xr-x  2 root   sys      4096 Feb 13 00:39 .
drwxr-xr-x  3 root   root     4096 Jan  1 1970 ..
-rwxr-xr-x  1 root   sys       417 Oct 12 2010 myconfig.old
-rwxr-xr-x  1 root   sys       417 Oct 12 2010 defaultconfig.novcs
-rwxr-xr-x  1 root   sys       697 Oct 12 2010 defaultconfig.vcs
-rw-r--r--  1 root   root     6800 Feb 13 00:37 startup-config
```

## Related Commands

[copy](#), [delete](#), [dir](#), [show file](#)

# rename (Access Gateway mode)

Provides a name for a port group or renames a port group in Access Gateway mode.

## Syntax

```
rename pg_name
```

## Parameters

*pg\_name*  
Port group name

## Modes

Port Grouping configuration mode

## Usage Guidelines

You must be in Port Grouping configuration mode for the specific port group to use this command. The *pg\_name* cannot exceed 64 characters.

## Examples

Renaming port group pg-1 to pg-array24.

```
device# configure terminal
device(config)# rbridge-id 3
device(config-rbridge-id-3)# ag
device(config-rbridge-id-3-ag)# pg 1
device(config-rbridge-id-3-ag-pg-1)# rename pg-array24
```

## Related Commands

[pg](#), [show ag pg](#)

# resequence access-list

Reassigns sequence numbers to entries of an existing MAC, IPv4, or IPv6 access list.

## Syntax

```
resequence access-list { ip | ipv6 | mac } name seq_num increment
```

## Parameters

**ip | ipv6 | mac**

Specifies the Layer 2 or Layer 3 ACL bound to an interface.

*name*

Specifies the name of a standard or an extended ACL. A maximum of 63 characters is allowed.

*seq\_num*

Specifies the starting sequence number in the ACL. Valid values range from 1 through 4294967290.

*increment*

Specifies a value to increment the sequence number between rules. Valid values range from 1 through 4294967290.

## Modes

Privileged EXEC mode

## Usage Guidelines

Reordering the sequence numbers is useful when you need to insert rules into an existing ACL and there are not enough sequence numbers available. When all sequence numbers between rules are exhausted, this feature allows the reassigning of new sequence numbers to entries of an existing access list.

## Examples

The following example reorders the rules in a MAC ACL.

```
device# show running-config mac access-list test
!
mac access-list standard test
 seq 1 permit 0011.2222.3333
 seq 2 permit 0011.2222.4444
 seq 3 permit 0011.2222.5555
 seq 4 deny 0011.2222.6666
!
device# resequence access-list mac test 10 10

device# show running-config mac access-list test
!
mac access-list standard test
 seq 10 permit 0011.2222.3333
 seq 20 permit 0011.2222.4444
 seq 30 permit 0011.2222.5555
 seq 40 deny 0011.2222.6666
!
```

The following example reorders the rules in an IPv6 ACL.

```
device# show running-config ipv6 access-list distList
!
ipv6 access-list standard distList
 seq 10 deny 2001:125:132:35::/64
 seq 20 deny 2001:54:131::/64
 seq 30 deny 2001:5409:2004::/64
 seq 40 permit any!
device# resequence access-list ipv6 distList 100 100
device# show running-config ipv6 access-list distList
!
ipv6 access-list standard distList
 seq 100 deny 2001:125:132:35::/64
 seq 200 deny 2001:54:131::/64
 seq 300 deny 2001:5409:2004::/64
 seq 400 permit any
!
```

# reserved-vlan

Defines the range of 802.1Q VLANs that cannot be created by means of the **interface vlan** command.

## Syntax

```
reserved-vlan start-VLAN-ID end-VLAN-ID
```

## Command Default

VLANs 4087 through 4095 are reserved on the switch.

## Parameters

*start-VLAN-ID*

Valid values range from 1 through 4086.

*end-VLAN-ID*

Valid values range from 1 through 4086.

## Modes

Global configuration mode

## Usage Guidelines

### NOTE

This command does not apply to service or transport VFs in a Virtual Fabrics context (VLAN ID > 4095).

The end value must be greater than the start value. This command succeeds if there are no VLANs configured in the specified range. Otherwise, an error instructs you to delete the configured VLANs in the specified range, or provide a different range.

VLAN 1002 is still the default FCoE VLAN. VLAN 1002 cannot be part of the reserved VLAN range unless some other VLAN is created for FCoE.

This command does not require a switch reboot.

## Examples

To set the configuration revision to 1:

```
device# configure terminal
device(config)# reserved-vlan 1550 1650
```

## Related Commands

[show default-vlan](#), [interface vlan](#), [show running reserved-vlan](#)

# resource-monitor cpu enable

Enables the CPU utilization monitoring service.

## Syntax

```
resource-monitor cpu enable
no resource-monitor cpu enable
```

## Command Default

Enabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This is a node-specific command.

The **no** form of the command disables the CPU utilization monitoring. Default action is to set to generate RASlog when CPU usage exceeds the threshold, which is 90%.

## Examples

The following example disables the CPU utilization monitoring service.

```
device# configure terminal
device(config)# rbridge-id 22
device(config-rbridge-id-22)# no resource-monitor cpu enable
```

The following example re-enables the CPU utilization monitoring service if it has been disabled.

```
device# configure terminal
device(config)# rbridge-id 22
device(config-rbridge-id-22)# resource-monitor cpu enable
```

## History

Release version	Command history
7.1.0	This command was introduced.



# resource-monitor memory

Configures the memory utilization monitoring service.

## Syntax

```
resource-monitor memory action raslog [ enable ][ threshold threshold ]
resource-monitor memory enable [ action { raslog } ][ threshold threshold ]
resource-monitor memory threshold threshold [ action { raslog } ][ enable ]
no resource-monitor memory enable
```

## Command Default

Enabled.

## Parameters

### action

Specifies the action to be taken when memory usage goes below the threshold.

### raslog

Specifies that a RASlog SRM-1002 message is sent.

### enable

Enables the memory utilization monitoring service.

### threshold *threshold*

Sets the threshold for free low memory. Valid values range from 50 through 200 MB of free low memory left. By default, the threshold is 100MB.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This is a node-specific command.

The **no** form of the command disables the memory monitoring on CPU.

## Examples

The following example specifies that a RASlog message is sent if the system memory is running low and goes beneath the threshold of 80 MB.

```
device# configure terminal
device(config)# rbridge-id 22
device(config-rbridge-id-22)# resource-monitor memory enable threshold 80 action raslog
```

## History

Release version	Command history
7.1.0	This command was introduced.

# resource-monitor process memory

Configures the per-process memory monitoring service.

## Syntax

```
resource-monitor process memory alarm alarm_threshold [ critical critical_threshold ] [ enable ]
resource-monitor process memory enable [ alarm alarm_threshold ] [ critical critical_threshold ]
resource-monitor process memory critical critical_threshold [ alarm alarm_threshold ] [ enable ]
no resource-monitor process memory enable
```

## Command Default

Enabled.

## Parameters

**alarm** *alarm\_threshold*

Specifies the alarm threshold, crossing which, specific RASlog is generated. Valid values range between 500 to 599 MB. The default is 500.

**enable**

Enables the pre-process memory monitoring service.

**critical** *critical\_threshold*

Specifies the critical threshold, crossing which, specific RASlog is generated. Valid values range between 600 to 699 MB. The default is 600.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This is a node-specific command. When the alarm threshold is reached, it generates the RASlog message SRM-1003. When the critical threshold is reached, it generates the RASlog message SRM-1004.

The **no** form of the command disables the pre-process memory monitoring on CPU.

## Examples

The following example enables the pre-process memory monitoring service and sets an alarm threshold of 550 MB and a critical threshold of 620 MB.

```
device# configure terminal
device (config)# rbridge-id 22
device(config-rbridge-id-22)# resource-monitor process memory enable alarm 550 critical 620
```

## History

Release version	Command history
7.1.0	This command was introduced.

# restrict-flooding

Restricts the flooding of egress BUM traffic from an AMPP port-profile port.

## Syntax

**restrict-flooding**

**no restrict-flooding**

## Command Default

Egress BUM traffic is allowed

## Modes

Port-profile configuration mode

## Usage Guidelines

This command is applicable only on the default profile and automatically applied to all the AMPP port-profile-ports on the device.

This command only blocks the egress BUM traffic. Ingress traffic, known as unicast traffic, is not impacted.

Use the **no restrict-flooding** version of this command to remove the restriction.

## Examples

To set the configuration revision to 1:

```
device# configure terminal
device(config)# interface tengigabitethernet 2/0/1
device(if-te-2/0/1)# port-profile-port
device(if-te-2/0/1)# restrict-flooding
```

# retain route-target all

Configures a route reflector (RR) to accept all route targets (RTs).

## Syntax

```
retain route-target all
no retain route-target all
```

## Command Default

Disabled.

## Modes

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

The **no** form of this command disables the retaining of all RTs.

## Examples

The following example configures a RR to accept all RTs.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# retain route-target all
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[address-family l2vpn evpn](#), [router bgp](#)

# revision

Assigns a version number to the Multiple Spanning Tree Protocol (MSTP) configuration.

## Syntax

**revision** *number*

**no revision**

## Command Default

The default is 0.

## Parameters

*number*

Specifies the revision or version number of the MSTP region. Valid values range from 0 through 255.

## Modes

]Spanning tree MSTP configuration mode

## Usage Guidelines

Enter **no revision** to return to the default setting.

If MSTP is enabled over VCS, this command must be executed on all RBridge nodes

## Examples

To set the configuration revision to 1:

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# revision 1
```

## Related Commands

[region](#), [show spanning-tree](#)

# rfc1583-compatibility (OSPF)

Configures compatibility with RFC 1583.

## Syntax

```
rfc1583-compatibility  
no rfc1583-compatibility
```

## Command Default

OSPF is compatible with RFC 1583 (OSPFv2).

## Modes

OSPF router configuration mode  
OSPF router VRF configuration mode

## Usage Guidelines

OSPF is compatible with RFC 1583 (OSPFv2) and maintains a single best route to an autonomous system (AS) boundary router in the OSPF routing table. Disabling this compatibility causes the OSPF routing table to maintain multiple intra-AS paths, which helps prevent routing loops.

Enter **no rfc1583-compatibility** to disable compatibility with RFC 1583.

## Examples

The following example disables compatibility with RFC 1583.

```
device# configure terminal  
device(config)# rbridge-id 5  
device(config-rbridge-id-5)# router ospf  
device(config-router-ospf-vrf-default-vrf)# no rfc1583-compatibility
```



# rfc1587-compatibility (OSPF)

Configures compatibility with RFC 1587.

## Syntax

```
rfc1587-compatibility  
no rfc1587-compatibility
```

## Command Default

OSPF is compatible with RFC 1587 (OSPFv2).

## Modes

OSPF configuration mode  
OSPF router VRF configuration mode

## Usage Guidelines

RFC 1587 is the original NSSA specification. Only part of the newer RFC 3101 is supported—the "no-summary" parameter and the handling of default-route LSAs when "no summary" is enabled.

Enter **no rfc1587-compatibility** to disable compatibility with RFC 1587.

## Examples

To disable compatibility with RFC 1587:

```
device# configure terminal  
device(config)# rbridge-id 5  
device(config-rbridge-id-5)#router ospf  
device(config-router-ospf-vrf-default-vrf)# no rfc1587-compatibility
```

# rib-route-limit

Limits the maximum number of BGP Routing Information Base (RIB) routes that can be installed in the Routing Table Manager (RTM).

## Syntax

```
rib-route-limit num
```

```
no rib-route-limit
```

## Command Default

This option is disabled. There is no limit.

## Parameters

*num*

Decimal value for the maximum number of RIB routes to be installed in the RTM. Range is from 1 through 65535.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

Beginning with Network OS v5.0.0, the **rib-route-limit** command controls the number of routes installed by BGP, irrespective of whether those BGP routes are the preferred routes in the system. BGP locally tracks the number of routes installed and the number of routes withdrawn from RIB. If the total number of routes installed exceeds the value specified by *num*, routes will not be installed.

If *num* is increased, route calculation is automatically triggered.

If *num* is decreased, the user is prompted to clear the BGP RTM.

## Examples

This example configures the device to limit the maximum number of BGP4 RIB routes that can be installed in the RTM.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# rib-route-limit 10000
```

This example configures the device to limit the maximum number of BGP4+ RIB routes that can be installed in the RTM in VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# rib-route-limit 32000
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family and to control the number of routes installed by BGP.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

# rmon alarm

Sets the RMON alarm conditions.

## Syntax

```
rmon alarm index snmp_oid interval seconds [ absolute | delta ] rising-threshold value event number [ falling-threshold value
event number [ owner name ]
```

```
no rmon alarm
```

## Command Default

No alarms are configured.

## Parameters

*index*

Specifies the RMON alarm index. Valid values range from 1 through 65535.

*snmp\_oid*

Specifies the MIB object to monitor. The variable must be in the SNMP OID format, for example, 1.3.6.1.2.1.16.1.1.1.5.65535. The object type must be a counter32.

**interval** *seconds*

Specifies the RMON alarm sample interval in seconds. Valid values range from 1 through 2147483648.

**absolute**

Sets the sample type as absolute.

**delta**

Sets the sample type as delta.

**rising-threshold** *value*

Specifies the RMON alarm rising threshold. Valid values range from 0 through 4294967295.

**event** *number*

Specifies the event for the rising alarm. Valid values range from 1 through 65535.

**falling-threshold** *value*

Specifies the RMON alarm falling threshold. Valid values range from 0 through 4294967295.

**event** *number*

Specifies the event for the rising alarm. Valid values range from 1 through 65535.

**owner** *name*

Specifies the identity of the owner. The maximum number of characters is 32.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no rmon alarm** to disable the alarm conditions.

## Examples

To set RMON alarm conditions:

```
device# configure terminal
device(config)# rmon alarm 100 1.3.6.1.2.1.16.1.1.1.5.65535 interval 5 absolute rising-threshold 10000
event 100 falling-threshold 1000 event 101 owner admin
```

## Related Commands

[rmon event](#), [show rmon](#)

# rmon collection history

Collects Ethernet group statistics for later retrieval.

## Syntax

```
rmon collection history number [ buckets bucket_number | interval seconds | owner name ]  
no rmon collection history number
```

## Command Default

RMON history collection is not enabled.

## Parameters

*number*

Specifies the RMON collection control index value. Valid values range from 1 through 65535.

**buckets** *bucket\_number*

Specifies the maximum number of buckets for the RMON collection history. Valid values range from 1 through 65535.

**interval** *seconds*

Specifies the alarm sample interval in seconds. Valid values range from 1 through 3600. The default value is 1800.

**owner** *name*

Specifies the identity of the owner. The maximum number of characters is 15.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command collects periodic statistical samples of Ethernet group statistics on a specific interface for later retrieval.

Enter **no rmon collection history** *number* to disable the history of statistics collection.

### NOTE

RMON configuration is not supported on breakout ports in Network OS versions prior to v6.0.0.

## Examples

To collect RMON statistics, with an RMON collection control index value of 5 for the owner named admin, on a specific Ethernet interface:

```
device# configure terminal  
device(config)# interface tengigabitethernet 170/0/1  
device(conf-if-te-170/0/1)# rmon collection history 5 owner admin
```

## Related Commands

[interface](#), [show rmon history](#)

# rmon collection stats

Collects Ethernet group statistics on a specific interface.

## Syntax

**rmon collection stats** *number* [ **owner name** ]

**no rmon collection stats** *number*

## Command Default

RMON statistic collection is not enabled.

## Parameters

*number*

Specifies the RMON collection control index value. Valid values range from 1 through 65535.

**owner name**

Specifies the identity of the owner.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no rmon collection stats** *number* to disable the collection of statistics.

Ethernet group statistics collection is not supported on ISL links.

### NOTE

RMON configuration is not supported on breakout ports in Network OS versions prior to v6.0.0.

## Examples

The following example shows how to collect RMON statistics, with an RMON collection control index value of 2 for the owner named admin, on a specific Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 170/0/1
device(conf-if-te-170/0/1)# rmon collection stats 2 owner admin
```

## Related Commands

[interface](#), [show rmon history](#)



# rmon event

Adds or removes an event in the RMON event table associated to the RMON alarm number.

## Syntax

```
rmon event index [ description word | log | owner name | trap word ]  
no rmon event
```

## Command Default

No events are configured.

## Parameters

*index*  
Specifies the RMON event number. Valid values range from 1 through 65535.

*description word*  
Specifies a description of the event.

log  
Generates an RMON log when an event is triggered.

*owner name*  
Specifies the owner of the event. The *name* string must be between 1 and 32 characters in length.

*trap word*  
Specifies the SNMP community or string name to identify this trap.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no rmon event** to remove the event configuration.

## Examples

To configure an RMON event:

```
device# configure terminal  
device(config)# rmon event 2 log description "My Errorstoday" owner gjack
```

## Related Commands

[show rmon history](#)

# role name

Creates or modifies a non-default role.

## Syntax

**role name** *role\_name* [ **desc** *description* ]

**no role name** *role\_name* [ **desc** *description* ]

## Parameters

*role\_name*

Specifies the name of the role.

**desc** *description*

Specifies an optional role description.

## Modes

Global configuration mode

## Usage Guidelines

For each role that you create, you define one or more rules. Each user is associated with one—and only one—role.

Role names are from 4 through 32 characters, must begin with a letter, and can contain alphanumeric characters and underscores. The name cannot be same as that of an existing user.

The description field supports up to 64 characters and can include any printable ASCII character, except for the following characters: single quotation mark ('), double quotation mark ("), exclamation point (!), colon (:), and semi-colon (;). If the description contains spaces, enclose the text in double quotation marks.

The maximum number of roles supported is 64, including the user and admin default roles.

To delete a role description, enter **no role name** *role\_name* **desc**.

To delete a role, enter **no role name** *role\_name*.

## Examples

The following example creates a role.

```
device# configure terminal
device(config)# role name tempAdmin desc "Daily admin functions"
```

The following example resets the description to the default value (no description).

```
device# configure terminal
device(config)# no role name tempAdmin desc
```

The following example deletes the role.

```
device# configure terminal
device(config)# no role name tempAdmin
```

## Related Commands

rule, show running-config role, show running-config rule

# root access console

Restricts the root access to the device to the console only.

## Syntax

**root access console**

**no root access console**

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no root access console** allows root access to the device through all terminals (SSH, Telnet, and console).

## Examples

Typical command output:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# do show running-config rbridge-id | include root
% No entries found.
device(config-rbridge-id-1)# root access console
device(config-rbridge-id-1)# do show running-config rbridge-id | include root
root access console
device(config-rbridge-id-1)#
```

## History

Release version	Command history
5.0.1a	This command was introduced.

## Related Commands

[root enable](#)

# root enable

Enables root access to the device following a firmware configuration.

## Syntax

**root enable**

**no root enable**

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no root enable** command disables root access to the device.

## Examples

Typical command output:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# do show running-config rbridge-id | include root
% No entries found.
device(config-rbridge-id-1)# root enable
% Info: Root password is at system default, for better security, you may want to change it.
device(config-rbridge-id-1)# do show running-config rbridge-id | include root
root enable
device(config-rbridge-id-1)#
```

## History

Release version	Command history
5.0.1a	This command was introduced.

## Related Commands

[root access console](#)

# route-map

Creates or deletes a route map instance, with a variety of options.

## Syntax

```
route-map name { permit | deny } instance_number
no route-map name [ permit | deny ] instance_number
```

## Parameters

*name*

The name of the route map. The string must be between 1 and 63 ASCII characters in length.

**permit**

Allows a matching pattern

**deny**

Disallows a matching pattern.

*instance\_number*

The instance ID. The range is from 1 through 65535.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This command is used in conjunction with the on **match** and **set** commands. For details on these commands, refer to the Network OS Command Reference for more information.

The maximum number of OSPF networks that can be advertised and processed in a single area in a router is limited to 600.

Enter **no route-map** *name* to remove the route-map name.

In a related note, the **continue** command configures the route map to continue to evaluate and execute match statements after a successful match occurs. The continue statement proceeds to the route map with the specified sequence number. If no sequence number is specified, the statement proceeds to the route map with the next sequence number (as an "implied" continue).

## Examples

This example configures a route map that allows a matching pattern.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# route-map test permit 5
```

To configure continue statements in a route map:

```
device(config)# rbridge-id 5
device(config-rbridge-id-5)# route-map mcontroutemap1 permit 1
device(config-routemap-mycontroutemap/permit/1)# match metric 10
device(config-routemap-mycontroutemap/permit/1)# set weight 10
device(config-routemap-mycontroutemap/permit/1)# match metric 10
device(config-routemap-mycontroutemap/permit/1)# continue 2
device(config-routemap-mycontroutemap/permit/1)# route-map mcontroutemap1 permit 2
device(config-routemap-mycontroutemap/permit/2)# match tag 10
device(config-routemap-mycontroutemap/permit/2)# set weight 20
```

# router bgp

Enables BGP routing.

## Syntax

`router bgp`

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no** form of this command to disable BGP routing.

## Examples

This example enables BGP routing.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)#
```



# router fabric-virtual-gateway

Configures the Fabric-Virtual-Gateway feature globally in VCS.

## Syntax

```
router fabric-virtual-gateway
```

```
no router fabric-virtual-gateway
```

## Command Default

None

## Modes

Global configuration mode

## Usage Guidelines

Enter the **no** form of the command to disable the Fabric-Virtual-Gateway configuration.

## Examples

The following example shows how to configure the Fabric-Virtual-Gateway feature globally and enable IPv4 address-family mode.

```
device(config)# router fabric-virtual-gateway
device(conf-router-fabric-virtual-gateway)# address-family ipv4
device(conf-address-family-ipv4)#
```

## History

Release version	Command history
5.0.1	This command was introduced.

# router ospf

Enables and configures the Open Shortest Path First version 2 (OSPFv2) routing protocol.

## Syntax

```
router ospf [ vrf name ]  
no router ospf
```

## Parameters

**vrf name**  
Specifies a nondefault VRF.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use this command to enable the OSPFv2 routing protocol and enter OSPF router or OSPF router VRF configuration mode. OSPFv2 maintains multiple instances of the routing protocol to exchange route information among various VRF instances.

The **no** form of the command deletes all current OSPF configuration and blocks any further OSPFv2 configuration.

## Examples

The following example enables OSPFv2 on a default VRF and enters OSPF VRF router configuration mode.

```
device# configure terminal  
device(config)# rbridge-id 5  
device(config-rbridge-id-5)# router ospf  
device(config-router-ospf-vrf-default-vrf)
```

The following example enables OSPFv2 on a non-default VRF and enters OSPF VRF router configuration mode.

```
device# configure terminal  
device(config)# rbridge-id 5  
device(config-rbridge-id-5)#router ospf vrf red  
device(config-router-ospf-vrf-red)
```

# router pim

Enables or disables the Protocol Independent Multicast (PIM) routing protocol.

## Syntax

```
router pim
```

```
no router pim
```

## Command Default

The PIM protocol is disabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This command launches the PIM router configuration mode.

Enter **exit** to exit this mode.

## Examples

To enable the PIM protocol:

```
device# configure terminal
device(config)# rbridge-id 128
device(config-rbridge-id-128)# router pim
```

# route-target (EVPN)

Enables auto-generation of the import and export route-target community attributes for an Ethernet Virtual Private Network (EVPN) instance.

## Syntax

```
route-target { both | import } auto [ ignore-as ]
route-target export auto
no route-target { both | import } auto [ ignore-as ]
no route-target export auto
```

## Command Default

Enabled.

## Parameters

### **both auto**

Specifies auto-generation of the import and export route-target community attributes.

### **ignore-as**

Specifies that the autonomous system (AS) number be ignored.

### **export auto**

Specifies auto-generation of the export route-target community attribute.

### **import auto**

Specifies auto-generation of the import route-target community attribute.

## Modes

EVPN instance configuration mode

## Usage Guidelines

The **no** form of this command removes configured route target parameters.

## Examples

The following example configures auto-generation of the import and export route-target community attributes for EVPN instance "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# route-target both auto
```

The following example configures auto-generation of the import route-target community attribute and specifies that the AS path be ignored to the route for EVPN instance "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# route-target import auto ignore-as
```

The following example configures auto-generation of the export route-target community attribute for EVPN instance "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# route-target export auto
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[evpn-instance](#)

# route-target (VNI)

Imports or exports the routes for a virtual network identifier (VNI) under an EVPN instance.

## Syntax

`route-target { both | export | import } admin-value:arbitrary-value`

`no route-target { both | export | import } admin-value:arbitrary-value`

## Command Default

Auto-generation of the import and export route-target community attributes is enabled.

## Parameters

### **both**

Specifies both the import and export route-target community attributes.

### **export**

Specifies the export route-target community attribute.

### **import**

Specifies the import route-target community attribute.

### *admin-value*

The administrative number assigned to the route. This can be a local ASN number or an IP address. The ASN number can be either a 2 byte (from 0 through 65535) or a 4 byte number from 0 through 4294967295).

### *arbitrary-value*

An arbitrary number you choose. The range of valid values is from 0 through 65535 if the ASN is 2 byte, or from 0 through 4294967295 if the ASN is 4 byte.

## Modes

VNI configuration mode

## Usage Guidelines

The **no** form of this command removes configured VNI route target parameters.

## Examples

The following example specifies both the export and import route-target community attributes and assigns the ip address 10.1.1.1 to the route for VNI 100 under EVPN instance "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# vni add 100
device(config-evpn-instance-myinstance)# vni 100
device(evpn-vni-100)# route-target both 10.1.1.1:1
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[evpn-instance](#), [vni \(EVPN\)](#)

## route-target (VRF)

Imports the routes to the VRF routing table from the BGP EVPN table when the import route target (RT) matches, and exports the routes from the VRF routing table to the BGP EVPN table with the configured export Route Target.

### Syntax

```
route-target { both | export | import } admin-value:arbitrary-value evpn
no route-target { both | export | import } admin-value:arbitrary-value evpn
```

### Command Default

Enabled.

### Parameters

#### **both**

Specifies both the import and export route-target community attributes.

#### **export**

Specifies the export route-target community attribute.

#### **import**

Specifies the import route-target community attribute.

#### *admin-value*

The administrative number assigned to the route. This can be a local ASN number or an IP address. The ASN number can be either a 2 byte (from 0 through 65535) or a 4 byte number from 0 through 4294967295).

#### *arbitrary-value*

An arbitrary number you choose. The range of valid values is from 0 through 65535 if the ASN is 2 byte, or from 0 through 4294967295 if the ASN is 4 byte.

#### **evpn**

Specifies the Ethernet Virtual Private Network (EVPN) route target.

### Modes

IPv4 address-family VRF configuration mode

IPv6 address-family VRF configuration mode

### Usage Guidelines

The **no** form of the command removes configured VRF route target parameters.



## Examples

The following example specifies the export route-target community attribute and assigns the local ASN number 32767:123 to the route. The EVPN route target is specified.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf green
device(config-vrf-green)# address-family ipv4 unicast
device(vrf-green-ipv4-unicast)# route-target export 32767:123 evpn
```

The following example specifies both the export and import route-target community attributes and assigns the ip address 10.1.1.1 to the route. The EVPN route target is specified.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf green
device(config-vrf-green)# address-family ipv4 unicast
device(vrf-green-ipv4-unicast)# route-target both 10.1.1.1:1 evpn
```

## History

Release version	Command history
7.0.0	The <b>evpn</b> keyword was added.

## Related Commands

[address-family unicast \(VRF\)](#), [vrf](#)

# rp-address

Adds or removes a static rendezvous-point (RP) address for a protocol-independent multicast (PIM) domain. You can also specify the name of a prefix list that defines a multicast-group range for which this RP hashes.

## Syntax

```
rp-address A.B.C.D [ prefix-list ]
```

```
no rp-address A.B.C.D
```

## Command Default

No interface is configured as the RP.

## Parameters

*A.B.C.D*

Specifies the IP address of the RP router.

*prefix-list*

Specifies the name of a prefix list defined by the **ip prefix-list** command. Permitted values are between 1 and 63 characters. Although the first character must be alphabetic, the others can be alphanumeric, underscores (\_), or minus signs (-).

## Modes

PIM router configuration mode

## Usage Guidelines

If a prefix list is not specified, the address is interpreted as the RP for the 224.0.0.0/4 address range.

To remove an RP address, enter the **no rp-address** command, specifying the IP address of that RP (but not a prefix-list).

## Examples

The following example sets the RP as the router at IP address 12.12.12.12, without specifying a prefix list.

```
device# configure terminal
device(config)# rbridge 1
device(config-rbridge-id-1)# router pim
device(config-pim-router)# rp-address 12.12.12.12
device(config-pim-router)#
```

The following example sets the RP as the router at IP address 12.12.12.12, specifying a prefix list. Any multicast-group address matching the rules in prefix-list "abc" is served by this RP.

```
device# configure terminal
device(config)# rbridge 1
device(config-rbridge-id-1)# router pim
device(config-pim-router)# rp-address 12.12.12.12 abc
device(config-pim-router)#
```

## Related Commands

[ip prefix-list](#), [router pim](#)

# rp-adv-interval

Configures the interval at which the candidate rendezvous point (RP) configured on the device sends candidate-RP advertisement messages to the bootstrap router (BSR).

## Syntax

`rp-adv-interval seconds`

`no rp-adv-interval seconds`

## Command Default

The device sends candidate-RP advertisement messages every 60 seconds.

## Parameters

*seconds*

Specifies the interval, in seconds, between advertisement messages. The range is 10 through 65535 seconds. The default is 60 seconds.

## Modes

PIM router configuration mode

PIM router VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the candidate-RP advertisement-message interval to 60 seconds.

## Examples

The following example configures the candidate-RP advertisement-message interval to 90 seconds.

```
device(config)# router pim
device(conf-pim-router)# rp-adv-interval 90
```

## History

Release version	Command history
7.1.0	This command was introduced.

# rp-candidate

Configures a device as a candidate rendezvous point (RP) for all multicast groups with the prefix 224.0.0.0/4, by default, and explicitly adds or deletes groups with other prefixes.

## Syntax

```
rp-candidate [ group-range A.B.C.D | interface <N>gigabitethernet ]
```

## Command Default

The PIM router is not available for selection as an RP.

## Parameters

**group-range** *A.B.C.D*

Specifies the group prefix IP address.

**interface** *<N>gigabitethernet*

Specifies the interface type.

## Modes

Router PIM configuration mode

## Usage Guidelines

The **no rp-candidate** command makes the PIM router cease to act as a candidate RP.

Configuring the **rp-candidate** command on an Ethernet, loopback, virtual, or tunnel interface, configures the device as a candidate RP for all multicast groups with the prefix 224.0.0.0/4, by default. You can configure the **rp-candidate add** command to add to those a group address or range of group addresses. You can configure the **rp-candidate delete** command to delete a group address or range of group addresses that were added to the default addresses.

### NOTE

You cannot delete the default group prefix.

The RP is the meeting point for PIM Sparse sources and receivers. A PIM Sparse domain can have multiple RPs, but each PIM Sparse multicast group address can have only one active RP. PIM Sparse routers learn the addresses of RPs and the groups for which they are responsible from messages that the bootstrap router (BSR) sends to each of the PIM Sparse routers.

Although you can configure the device as only a candidate BSR or an RP, it is recommended that you configure the same interface on the same device as both a BSR and an RP.

## Examples

This example adds a multicast group range.

```
device# configure terminal
device(config)# rbridge 1
device(config-rbridge-id-1)# router pim
device(conf-pim-router)# rp-candidate group-range 230.1.0.0 16
```

This example adds an interface acting as an RP candidate.

```
device# configure terminal
device(config)# rbridge 1
device(config-rbridge-id-1)# router pim
device(conf-pim-router)# rp-candidate interface ve 101 priority 10
```

# rpf-mode

Enables strict or loose unicast Reverse Path Forwarding (uRPF) mode on an interface.

## Syntax

```
rpf-mode { loose | strict }  
no rpf-mode
```

## Command Default

uRPF mode is not enabled.

## Parameters

### strict

Specifies uRPF strict mode. For details, refer to the Usage Guidelines.

### loose

Specifies uRPF loose mode. For details, refer to the Usage Guidelines.

## Modes

Interface configuration mode

## Usage Guidelines

This command is applicable only on Layer 3 physical, port-channel, and VE interfaces.

Loose mode permits a packet if the source address matches a routing table entry. Packets are dropped only if the source address is not reachable through any device interface.

Strict mode requires that a packet matches a known route entry—as described in loose mode—and also that it arrives at the interface as described in the router table next-hop information. Packets that do not match both of these criteria are dropped.

Both loose mode and strict mode include the default route in the Source IP (SIP) lookup.

The **no** form of the command disables uRPF on the interface.

## Examples

The following example enables uRPF on an interface, in strict mode.

```
device# configure terminal  
device(config)# interface ten 1/0/3  
device(conf-if-te-1/0/3)# rpf-mode strict
```

## History

Release version	Command history
7.2.0	This command was introduced.



# rspan-vlan

Configures the VLAN to support RSPAN (Remote Switched Port Analyzer) traffic analysis.

## Syntax

```
rspan-vlan
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

RSPAN extends SPAN by enabling remote monitoring of multiple switches across your network.

All participating devices must be trunk-connected at Layer 2, and RSPAN must be configured on all the switches participating in the RSPAN session.

## Examples

Typical execution of this command.

```
device# configure terminal
device(config)# interface vlan 300
device(config-vlan-300)# rspan-vlan
```

## Related Commands

[interface vlan](#), [show vlan](#)

# rule

Creates role-based access permissions (RBAC) associated with a role.

## Syntax

```
rule index [ action { accept | reject } ] [ operation { read-only | read-write } ] role role_name command command_name
no rule index
```

## Command Default

The default for **action** is **accept**. The default for **operation** is **read-write**.

## Parameters

*index*

Specifies a numeric identifier for the rule. Valid values range from 1 through 512.

**action** **accept** | **reject**

(Optional) Specifies whether the user is accepted or rejected while attempting to execute the specified command. The default value is **accept**.

**operation** **read-only** | **read-write**

(Optional) Specifies the type of operation permitted. The default value is **read-write**.

**role** *role\_name*

Specifies the name of the role for which the rule is defined.

**command** *command\_name*

Specifies the command for which access is defined. Separate commands with a space. To display a list of supported commands, type a question mark (?).

## Modes

Global configuration mode

## Usage Guidelines

For each role that you create, you define one or more rules. Each account is associated with one—and only one—role.

When you create a rule, the *index*, **role**, and **command** operands are mandatory; the **action** and **operation** operands are optional.

The maximum number of rules is 512.

When you modify a rule, all operands except *index* and **role** are optional.

Enter **no rule** *index* to remove the specified rule.

## Examples

The following example creates rules enabling the NetworkSecurityAdmin role to create user accounts.

```
device# configure terminal
device(config)# rule 150 action accept operation read-write role NetworkSecurityAdmin command config
device(config)# rule 155 action accept operation read-write role NetworkSecurityAdmin command username
```

The following example deletes a rule.

```
device# configure terminal
device(config)# no rule 155
```

## Related Commands

[role name](#), [show running-config role](#), [show running-config rule](#)

# rule (MAPS)

Creates and modifies user-defined rules for Monitoring and Alerting Policy Suite (MAPS).

## Syntax

```
rule { rule_name group group_name monitor monitor_name interval { none | min | hour | day } op { gt | lt | le | ge | eq } value
      threshold_value }
```

```
no rule { rule_name }
```

## Command Default

By default, only pre-defined rules are available.

## Parameters

*rule\_name*

The name for this user-defined rule. The

**group** *group\_name*

The name of the logical group of ports to which the rule is applied.

**monitor** *monitor\_name*

The monitor name to which the rule is applied.

**interval** { **none** | **min** | **hour** | **day** }

The **interval** keyword defines how often the rule is executed. The time values for the interval keywords are defined in the following list:

- **none**: There is no interval. The rule is always applied.
- **min**: The response is triggered if the rule is broken once within the last 60 seconds.
- **hour**: The response is triggered if the rule is broken once within the last 60 minutes.
- **day**: The response is triggered if the rule is broken once within the last 24 hours.

**op** { **gt** | **lt** | **le** | **ge** | **eq** }

The **op** keyword is defined as the mathematical operator for the rule. The other keywords are defined in the following list:

- **gt**: Stands for the "greater than" symbol ( > ).
- **lt**: Stands for the "less than" symbol ( < ).
- **ge**: Stands for the "greater than or equal to" symbol ( >= ).
- **le**: Stands for the "less than or equal to" symbol ( <= ).
- **eq**: Stands for the "equals" symbol ( = ).

**value** *threshold\_value*

The value at which the operator is triggered.

## Modes

MAPS configuration mode

## Usage Guidelines

The administrator can configure the following rules:

- Simple rules that monitor a single counter in each rule.
- Multiple rules with respective conditions defined across multiple time intervals (hour, minute, or day) for the same counter (the rules can be active simultaneously).
- Different rules for the same counter and time interval but with different thresholds (the rules can be active simultaneously).

This provides the capability to create a rule for any kind of scenario that is needed.

The valid monitor names are:

- CRCALN
- RX\_SYM\_ERR
- RX\_ABN\_FRAME
- ASIC\_PKTDROP
- SEC\_TELNET
- SEC\_LV
- TEMP
- CURRENT
- VOLTAGE
- RXP
- TXP
- FLASH\_USAGE
- MEMORY\_USAGE
- CPU
- BAD\_TEMP
- BAD\_PWR
- BAD\_FAN
- SFP\_STATE
- PS\_STATE
- FAN\_STATE
- ETH\_MGMT\_PORT\_STATE
- SFP\_TEMP
- WWN\_DOWN
- DOWN\_SFM
- FAULTY\_BLADE
- HA\_SYNC
- BLADE\_STATE
- WWN

The **no rule** of the command deletes the rule.

## Examples

Typical command example:

```
device#configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# maps
device(config-rbridge-id-1-maps)# rule RuleOne group EthUser monitor BAD_TEMP interval day op eq value 6
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.2.0	Monitor name added.

## Related Commands

[group, policy](#)

# run-mode

For an implementation of an event-handler profile on an RBridge, specifies if an event-handler action is run in exclusive or non-exclusive mode. The default non-exclusive mode enables cluster formation to run simultaneously with a triggered action.

## Syntax

**run-mode** *exclusivity-mode*

**no run-mode**

## Command Default

The default mode is **non-exclusive**.

## Parameters

*exclusivity-mode*

Specifies if a triggered event-handler action is run in exclusive or non-exclusive mode. The default setting is **non-exclusive**.

**exclusive**

From the triggering of an event-handler action through the completion of the action, cluster formation is not allowed to run.

**non-exclusive**

Cluster formation can occur while a triggered action is in progress.

## Modes

Event-handler activation mode

## Usage Guidelines

The **no** form of this command resets the **run-mode** setting to the default **non-exclusive** option.

## Examples

The following example sets the run-mode to **exclusive**.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# run-mode exclusive
```

The following example resets **run-mode** to the default value of **non-exclusive**.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# no run-mode
```

## History

Release version	Command history
6.0.1	This command was introduced.



# scheduler

Specifies the scheduling attributes along with the TC shape rate.

## Syntax

```
scheduler sp_count [ shape_rate | [ shape_rate ... shape_rate ] dwrr [ weight | weight ... weight ]
```

## Parameters

*sp\_count*

Specifies how many strict priority queues for each port scheduler. The range of valid values is from 0 through 8.

*shape\_rate*

Specifies the shape rate on strict priority queues. The range of valid values are from 28000 kbps to the maximum interface speed.

**dwrr** *weight*

Specifies the dwrr weight percentage for the queue. The range of valid values is from 1% through 100%, and the sum of all dwrr weights should not exceed 100%.

## Modes

Policy-map configuration mode

## Usage Guidelines

There are total eight queues are present on an interface. The number of dwrr queues present depends on the SP\_COUNT value. For example if the SP\_COUNT is two, then there are two strict priority queues and six dwrr queues.

This command is allowed only for the Egress direction.

This command can only be configured in for the **class class-default** command.

This command is mutually exclusive of the **port-shape** and **police** commands.

## Examples

Typical command example:

```
device# configure terminal
device(config)#policy-map mutation
device(config-policymap)#class class-default
device(config-policyclass)# scheduler 3 31000 32000 33000 dwrr 20 20 20 10 10
```

## Related Commands

[class](#), [policy-map](#)

# script reload

Reloads scripts from disk and shows a variety of script-related information.

## Syntax

```
script reload [ all [ debug ] ] | debug | diff [ debug ] | errors [ debug ] ]
```

## Command Default

None

## Parameters

### all

Displays information about all scripts.

### debug

Displays additional debug information about the scripts.

### diff

Displays information about scripts that have changed since the last reload.

### errors

Displays information about erroneous scripts.

## Modes

Privileged EXEC mode

## Examples

To reload scripts from disk:

```
device# script reload
```

## History

Release version	Command history
5.0.1	This command was introduced.

# scp

Imports a TLS certificate and private key pair.

## Syntax

```
scp { tlscert_filename scpuser@ip_address:flash-tlscert }
```

```
scp { tlsprivkey_filename scpuser@ip_address:flash-tlsprivkey }
```

## Command Default

The certificates are not installed on the device.

## Parameters

*tlscert\_filename*

The TLS certificate filename.

**scpuser@ip\_address:flash-tlscert**

The address for the TLS certificate server.

*tlsprivkey\_filename*

The TLS private key filename.

**scpuser@ip\_address:flash-tlsprivkey**

The address for the TLS key server.

## Modes

This is a Linux command.

## Usage Guidelines

The SSL/TLS protocol uses a pair of keys – one private, one public – to authenticate, secure and manage secure connections. These keys are created together as a pair and work together during the TLS handshake process to set up a secure session.

You must create a username named "scpuser" with admin privileges before you import the certificate or key.

Both the certificate and the key is used to establish secure connection over TLS by restarting the http server with the **http server use-vrf <VRF Name> shutdown** and **no http server use-vrf <VRF Name> shutdown** commands.

## Examples

Example of creating the "scpuser" role and importing the certificate and key.

```
device# configure terminal
device(config)# username scpuser role admin password 123456

# scp certificatefile scpuser@10.16.0.112:flash-tlscert <----Run from a Linux machine
# scp keyfile scpuser@10.16.0.112:flash-tlsprivkey <----Run from a Linux machine

device(config)# http server use-vrf myvrf shutdown
device(config)# no http server use-vrf myvrf shutdown
```

## History

Release version	Command history
7.0.2	This command was introduced.

# secpolicy activate

Activates the defined device connection control (SCC) policy and its member set.

## Syntax

```
secpolicy activate [ rbridge-id { rbridge-id | all } ]
```

## Command Default

Any device is allowed to join the fabric.

The SCC policy does not exist until it is defined and activated.

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

The SCC policy is used to restrict which switches can join the fabric by either accepting or rejecting the connection between two switches. Switches are checked against the policy each time an E\_Port-to-EX\_Port connection is made. The policy is named SCC\_POLICY and accepts device members listed as World Wide Names (WWNs).

Although the **active-policy** is listed under the possible completions of **secpolicy** command, the *defined-policy* parameter should be used to create or add policy or members. The **secpolicy activate** command activates the policy.

A defined SCC policy must exist before you can activate the policy. You create the policy with the **secpolicy defined-policy** command.

During configuration replay, the defined and active policies are replayed and the E\_Ports are enabled or disabled based on the SCC policy entries in the active policy list.

During **copy file running-config** command execution, only the defined policy in the device is updated with the config file entries; the active policy entries remain unchanged. In this case, you must use the **secpolicy activate** command to activate the defined policy list.

This command is supported on the Extreme VDX 6740, VDX 2741, and VDX 2746.

secpolicy activate

## Examples

```
device# secpolicy activate rbridge-id 3
```

## Related Commands

[secpolicy defined-policy](#)

# secpolicy defined-policy

Creates the device connection control (SCC) policy and adds the SCC defined policy set members (WWNs).

## Syntax

```
secpolicy defined-policy SCC_POLICY [ member-entry device_wwn ]
```

```
no secpolicy defined-policy SCC_POLICY [ member-entry device_wwn ]
```

## Command Default

Any device is allowed to join the fabric. The SCC policy does not exist until it is created.

## Parameters

**member-entry** *device\_wwn*

The device WWN to be added to the SCC defined policy set.

## Modes

Global configuration mode

RBridge ID configuration mode

## Usage Guidelines

The SCC policy defines which switches can join the fabric by either accepting or rejecting the connection between two switches. switches are checked against the policy each time an E\_Port-to-EX\_Port connection is made. The policy is named SCC\_POLICY and accepts members listed as WWNs.

When you execute this command, the SCC policy entry is created (if not present) and the WWNs are added to the SCC policy.

This command is not distributed across the cluster. The RBridge ID of the node should be used to configure policy configurations.

Although the **active-policy** is listed under the possible completions of **secpolicy** command, the **defined-policy** parameter should be used to create or add policy or members. Entering **secpolicy activate** activates the policy.

You can add multiple WWNs separated by a comma.

After you configure the defined SCC\_POLICY, run **secpolicy activate** to apply the changes to the active policy set.

This command is supported on the Extreme VDX 6740, VDX 2741, and VDX 2746.

Enter **no secpolicy defined-policy SCC\_POLICY member-entry device\_wwn** to remove a device (WWN) from the defined policy member set.

Enter **no secpolicy defined-policy SCC\_POLICY** to remove the SCC policy, along with all of the defined policy members.

## Examples

To create the defined SCC policy:

```
device# configure terminal
device(config)# rbridge-id 3
device(config-rbridge-id-3)# secpolicy defined-policy SCC_POLICY
device(config-defined-policy-SCC_POLICY)# exit
```

To add a device WWN to the policy set:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# secpolicy defined-policy SCC_POLICY member-entry 10:00:00:05:1e:
00:69:00
device(config-member-entry-10:00:00:05:1e:00:69:00)# exit
device(config-defined-policy-SCC_POLICY)# exit
device(config-rbridge-id-3)# exit
```

To remove an entry from the policy list of rbridge id 3:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# no secpolicy defined-policy SCC_POLICY member-entry 10:00:00:05:1e:00:69:01
```

To remove the SCC\_POLICY entry of rbridge id 3:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# no secpolicy defined-policy SCC_POLICY
```

## Related Commands

[secpolicy activate](#)



# security-profile (AMPP)

Activates the security-profile mode for AMPP.

## Syntax

**security-profile**

**no security-profile**

## Modes

Port-profile configuration mode

## Usage Guidelines

The security-profile mode for AMPP allows configuration of security attributes of a port-profile.

Enter **no security-profile** to remove the profile.

## Examples

To activate the security-profile mode for AMPP:

```
device# configure terminal
device(config)# port-profile sample-profile
device(conf-pp)# security-profile
```

# seq (IPv4 extended ACLs)

Inserts filtering rules in IPv4 extended ACLs. Extended ACLs permit or deny traffic according to source and destination addresses, as well as other parameters.

## Syntax

```
seq seq-value { permit | deny | hard-drop } ip-protocol { any | S_IPAddress mask | host S_IPAddress } [ source-operator [ S_port-numbers ] ] { any | D_IPAddress mask | host D_IPAddress } [ destination-operator [ D_port-numbers ] ] [ vlan vlanID ] [ dscp DSCPvalue ] [ TCP-flags ] [ count ] [ log ]

{ permit | deny | hard-drop } ip-protocol { any | S_IPAddress mask | host S_IPAddress } [ source-operator [ S_port-numbers ] ] { any | D_IPAddress mask | host D_IPAddress } [ destination-operator [ D_port-numbers ] ] [ vlan vlanID ] [ dscp DSCPvalue ] [ TCP-flags ] [ count ] [ log ]

no seq seq-value

no { permit | deny | hard-drop } ip-protocol { any | S_IPAddress mask | host S_IPAddress } [ source-operator [ S_port-numbers ] ] { any | D_IPAddress mask | host D_IPAddress } [ destination-operator [ D_port-numbers ] ] [ vlan vlanID ] [ dscp DSCPvalue ] [ TCP-flags ] [ count ] [ log ]
```

## Parameters

### seq

(Optional) Enables you to assign a sequence number to the rule. If you do not specify **seq seq-value**, the rule is added at the end of the list.

*seq-value*

Valid values range from 0 through 4294967290.

### permit

Specifies rules to permit traffic.

### deny

Specifies rules to deny traffic.

### hard-drop

Overrides the trap behavior for control frames and data frames such as echo request (ping). However, **hard-drop** does not override a **permit** for this address in a preceding rule.

### ip-protocol

Indicates the type of IP packet you are filtering. The options are as follows:

<0-255>

Protocol number custom value from 0 through 255.

### icmp

Internet Control Message Protocol

### ip

Any IP protocol

### tcp

Transmission Control Protocol

**udp**  
User Datagram Protocol

**any**  
Specifies all source addresses.

*S\_IPAddress*  
Specifies a source address for which you want to filter the sub-net.

*mask*  
Defines a mask, whose effect is to specify a subnet that includes the source address that you specified. For options to specify the mask, see the Usage Guidelines.

**host**  
Specifies a source address.

*S\_IPAddress*  
The source address.

*source-operator*  
The following operators are available:

**eq**  
The policy applies to the TCP or UDP port name or number you enter after **eq**.

**gt**  
The policy applies to TCP or UDP port numbers greater than the port number or the numeric equivalent of the port name you enter after **gt**.

**lt**  
The policy applies to TCP or UDP port numbers that are less than the port number or the numeric equivalent of the port name you enter after **lt**.

**neq**  
The policy applies to all TCP or UDP port numbers except the port number or port name you enter after **neq**.

**range**  
The policy applies to all TCP or UDP port numbers that are between the first TCP or UDP port name or number and the second one you enter following the **range** keyword. The range includes the port names or numbers you enter. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: **range 23 53** (two values separated by a space). The first port number in the range must be lower than the last number in the range.

*S\_port-numbers*  
(Valid only when *ip-protocol* is UDP or TCP) Specifies one or more source port numbers.

**any**  
Specifies all destination addresses.

*D\_IPAddress*  
Specifies a destination address for which you want to filter the sub-net.

*mask*  
Defines a mask, whose effect is to specify a subnet that includes the destination address that you specified. For options to specify the mask, see the Usage Guidelines.

**host**  
Specifies a destination address.

*D\_IPAddress*

The destination address.

*destination-operator*

The following operators are available:

**eq**

The policy applies to the TCP or UDP port name or number you enter after **eq**.

**gt**

The policy applies to TCP or UDP port numbers greater than the port number or the numeric equivalent of the port name you enter after **gt**.

**lt**

The policy applies to TCP or UDP port numbers that are less than the port number or the numeric equivalent of the port name you enter after **lt**.

**neq**

The policy applies to all TCP or UDP port numbers except the port number or port name you enter after **neq**.

**range**

The policy applies to all TCP or UDP port numbers that are between the first TCP or UDP port name or number and the second one you enter following the **range** keyword. The range includes the port names or numbers you enter. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: range 23 53. The first port number in the range must be lower than the last number in the range.

*D\_port\_numbers*

(Valid only when *ip-protocol* is UDP or TCP) The operator that you specified determines how *D\_port\_numbers* are applied.

**vlan** *vlanID*

Specifies a VLAN interface to which the ACL is bound.

**dscp**

Matches *DSCPvalue* against the DSCP value of the packet.

*DSCPvalue*

From 0 through 63.

*TCP-flags*

Under TCP, you can specify one or more of the following flags:

**ack**

Filters packets for which the **ack** (acknowledge) flag is set.

**fin**

Filters packets for which the **fin** (finish) flag is set.

**rst**

Filters packets for which the **rst** (reset) flag is set.

**sync**

Filters packets for which the **syn** (synchronize) flag is set.

**urg**

Filters packets for which the **urg** (urgent) flag is set.

**push**

Filters packets for which the **psh** (push) flag is set.

**count**

Enables statistics for the rule.

**log**

(Available for **permit** or **deny** only) Enables logging for the rule. In addition, the ACL log buffer must be enabled, using the **debug access-list-log buffer** command.

## Modes

ACL configuration mode

## Usage Guidelines

This command configures rules to permit or drop traffic based on source and destination addresses and protocol type. You can also enable counters and logging for specified rules.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list.

The hard-drop option can prevent trapping of control frames. As a result, it could interfere with normal operations of the protocols.

For an ACL applied to a management interface, **hard-drop** keywords are interpreted as **deny** keywords.

You can specify a mask in either of the following ways:

- Wildcard mask format—for example, 0.0.0.255. The advantage of this format is that it enables you mask any bit, for example by specifying 0.255.0.255.
- Classless Interdomain Routing (CIDR) format—in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 as wildcard mask format.

If you are defining rules for a QoS ACL, be aware of the following considerations for ACLs implemented under flow-based QoS. For more information, refer to the "Configuring QoS" > "Flow-based QoS" section of the *Network OS Layer 2 Switching Configuration Guide*.

- Do not include the **count** keyword in ACLs intended for flow-based QoS implementation, because such ACLs automatically share a common counter.
- The **deny** keyword functions as a PASS THROUGH: For a match, QoS action defined for that class is not applied.
- The **hard-drop** keyword is equivalent to the **deny** keyword.

For both interface ACLs and receive-path ACLs, you use identical commands to create the ACLs. You also use identical commands to define permit/deny rules in the ACLs. The only variance is the command you use to apply the ACL:

- To apply an interface ACL, from an interface-subtype configuration mode you use the { **ip** | **ipv6** | **mac** } **access-group** command.
- To apply a receive-path ACL, from RBridge ID configuration mode you use the { **ip** | **ipv6** } **receive access-group** command.

To delete a rule from an ACL, do the relevant of the following:

- If you know the rule number, enter **no seq seq-value**.

- If you do not know the rule number, type **no** followed by the full syntax except for **seq seq-value**.

## Examples

The following example creates an IPv4 extended ACL and defines rules:

```
device(config)# ip access-list extended extdACL5
device(config-ipacl-ext)# seq 5 deny tcp host 10.24.26.145 any eq 23
device(config-ipacl-ext)# seq 7 deny tcp any any eq 80
device(config-ipacl-ext)# seq 10 deny udp any any range 10 25
device(config-ipacl-ext)# seq 15 permit tcp any any
```

The following example creates an IPv4 extended ACL, defines rules in the ACL, and applies it as a receive-path ACL to an RBridge.

```
device(config)# ip access-list extended ipv4-receive-acl-example
device(conf-ipacl-ext)# hard-drop tcp host 10.0.0.1 any count
device(conf-ipacl-ext)# hard-drop udp any host 20.0.0.1 count
device(conf-ipacl-ext)# permit tcp host 10.0.0.2 any eq telnet count
device(conf-ipacl-ext)# permit tcp host 10.0.0.2 any eq bgp count
device(conf-ipacl-ext)# hard-drop tcp host 10.0.0.3 host 224.0.0.1 count

device(conf-ipacl-ext)# rb 1
device(config-rbridge-id-1)# ip receive access-group ipv4-receive-acl-example in
```

## Related Commands

[ip access-group](#), [ip access-list](#), [show running-config ip access-list](#)

# seq (IPv6 extended ACLs)

Inserts filtering rules in IPv6 extended ACLs. Extended ACLs permit or deny traffic according to source and destination addresses, as well as other parameters.

## Syntax

```
seq seq-value { permit | deny | hard-drop } ip-protocol { any | S_IPAddress / prefix_len | host S_IPAddress } [ source-operator [ S_port-numbers ] ] { any | D_IPAddress / prefix_len | host D_IPAddress } [ destination-operator [ D_port-numbers ] ] [ vlan vlanID ] [ dscp DSCPvalue ] [ TCP-flags ] [ count ] [ log ]

{ permit | deny | hard-drop } ip-protocol { any | S_IPAddress / prefix_len | host S_IPAddress } [ source-operator [ S_port-numbers ] ] { any | D_IPAddress / prefix_len | host D_IPAddress } [ destination-operator [ D_port-numbers ] ] [ vlan vlanID ] [ dscp DSCPvalue ] [ TCP-flags ] [ count ] [ log ]

no seq seq-value

no { permit | deny | hard-drop } ip-protocol { any | S_IPAddress / prefix_len | host S_IPAddress } [ source-operator [ S_port-numbers ] ] { any | D_IPAddress / prefix_len | host D_IPAddress } [ destination-operator [ D_port-numbers ] ] [ vlan vlanID ] [ dscp DSCPvalue ] [ TCP-flags ] [ count ] [ log ]
```

## Parameters

### seq

(Optional) Enables you to assign a sequence number to the rule. If you do not specify **seq seq-value**, the rule is added at the end of the list.

*seq-value*

Valid values range from 0 through 4294967290.

### permit

Specifies rules to permit traffic.

### deny

Specifies rules to deny traffic.

### hard-drop

Overrides the trap behavior for control frames and data frames such as echo request (ping). However, **hard-drop** does not override a **permit** for this address in a preceding rule.

### ip-protocol

Indicates the type of IP packet you are filtering. The options are as follows:

<0-255>

Protocol number custom value from 0 through 255.

### ipv6-icmp

Internet Control Message Protocol

### ipv6

Any IP protocol

### tcp

Transmission Control Protocol

**udp**  
User Datagram Protocol

**any**  
Specifies all source addresses.

*S\_IPAddress*  
Specifies a source address for which you want to filter the subnet. For options to abbreviate the address, see the Usage Guidelines.

*prefix\_len*  
Indicates how many of the high-order, contiguous bits of the address comprise the IPv6 prefix.

**host**  
Specifies a source address.

*S\_IPAddress*  
The specific address. For options to abbreviate the address, see the Usage Guidelines.

*source-operator*  
The following operators are available:

**eq**  
The policy applies to the TCP or UDP port name or number you enter after **eq**.

**gt**  
The policy applies to TCP or UDP port numbers equal to or greater than the port number or the numeric equivalent of the port name you enter after **gt**.

**lt**  
The policy applies to TCP or UDP port numbers that are equal to or less than the port number or the numeric equivalent of the port name you enter after **lt**.

**neq**  
The policy applies to all TCP or UDP port numbers except the port number or port name you enter after **neq**.

**range**  
The policy applies to all TCP or UDP port numbers that are between the first TCP or UDP port name or number and the second one you enter following the **range** keyword. The range includes the port names or numbers you enter. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: **range 23 53** (two values separated by a space). The first port number in the range must be lower than the last number in the range.

*S\_port-numbers*  
(Valid only when *ip-protocol* is UDP or TCP) Specify one or more port numbers.

**any**  
Specifies all destination addresses.

*D\_IPAddress*  
Specifies a destination address for which you want to filter the subnet. For options to abbreviate the address, see the Usage Guidelines.

*prefix\_len*  
Indicates how many of the high-order, contiguous bits of the address comprise the IPv6 prefix.

**host**  
Specifies a destination address.



*D\_IPaddress*

The destination address. For options to abbreviate the address, see the Usage Guidelines.

*destination-operator*

Specifies one of the following destination operators:

**eq**

The policy applies to the TCP or UDP port name or number you enter after **eq**.

**gt**

The policy applies to TCP or UDP port numbers equal to or greater than the port number or the numeric equivalent of the port name you enter after **gt**.

**lt**

The policy applies to TCP or UDP port numbers that are equal to or less than the port number or the numeric equivalent of the port name you enter after **lt**.

**neq**

The policy applies to all TCP or UDP port numbers except the port number or port name you enter after **neq**.

**range**

The policy applies to all TCP or UDP port numbers that are between the first TCP or UDP port name or number and the second one you enter following the **range** keyword. The range includes the port names or numbers you enter. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: **range 23 53**. The first port number in the range must be lower than the last number in the range.

*D\_port\_numbers*

(Valid only when *ip-protocol* is UDP or TCP) Specify one or more destination port numbers.

**vlan** *vlanID*

Specifies a VLAN interface to which the ACL is bound.

**dscp**

Matches *DSCPvalue* against the DSCP value of the packet.

*DSCPvalue*

From 0 through 63.

*TCP-flags*

Under TCP, you can specify one or more of the following flags:

**ack**

Filters packets for which the **ack** (acknowledge) flag is set.

**fin**

Filters packets for which the **fin** (finish) flag is set.

**rst**

Filters packets for which the **rst** (reset) flag is set.

**sync**

Filters packets for which the **syn** (synchronize) flag is set.

**urg**

Filters packets for which the **urg** (urgent) flag is set.

**push**

Filters packets for which the **psh** (push) flag is set.

**count**

Enables statistics for the rule.

**log**

(Available for **permit** or **deny** only) Enables logging for the rule. In addition, the ACL log buffer must be enabled, using the **debug access-list-log buffer** command.

## Modes

ACL configuration mode

## Usage Guidelines

This command configures rules to permit or drop traffic based on source and destination addresses and protocol type. You can also enable counters and logging for specified rules.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list.

The hard-drop option can prevent trapping of control frames. As a result, it could interfere with normal operations of the protocols.

For an ACL applied to a management interface, **hard-drop** keywords are interpreted as **deny** keywords.

You can abbreviate an IPv6 address by using one or more of the following rules:

- Remove one or more leading zeros from one or more groups of hexadecimal digits; this is usually done to either all or none of the leading zeros. (For example, convert the group 0042 to 42.)
- Omit consecutive sections of zeros, using a double colon (::) to denote the omitted sections. The double colon may only be used once in any given address, as the address would be indeterminate if the double colon were used multiple times. A double colon may not be used to denote an omitted single section of zeros. (For example, 2001:db8::1:2 is valid, but 2001:db8::1::2 or 2001:db8::1:1:1:1 are not permitted.)

On the VDX 8770, filtering of IPv6 traffic by DSCP value is supported for ingress only.

If you are defining rules for a QoS ACL, be aware of the following considerations for ACLs implemented under flow-based QoS. For more information, refer to the "Configuring QoS" > "Flow-based QoS" section of the *Network OS Layer 2 Switching Configuration Guide*.

- Do not include the **count** keyword in ACLs intended for flow-based QoS implementation, because such ACLs automatically share a common counter.
- The **deny** keyword functions as a PASS THROUGH: For a match, QoS action defined for that class is not applied.
- The **hard-drop** keyword is equivalent to the **deny** keyword.

To delete a rule from an ACL, do the relevant of the following:

- If you know the rule number, enter **no seq seq-value**.
- If you do not know the rule number, type **no** followed by the full syntax except for **seq seq-value**.

## Examples

The following example creates an IPv6 extended ACL, defines a rule for it, and applies the ACL to an interface.

```
device# configure
device(config)# ipv6 access-list extended ip_acl_1
device(conf-ip6acl-ext)# seq 10 deny ipv6 2001:2002:1234:1::/64 2001:1001:1234:1::/64 count
device(conf-ip6acl-ext)# exit
device(config)# interface ten 122/5/22
device(conf-if-te-122/5/22)# ipv6 access-group ip_acl_1 in
```

The following example creates an IPv6 extended ACL, defines rules in the ACL, and applies it as a receive-path ACL to an RBridge.

```
device(config)# ipv6 access-list extended ipv6-receive-acl-example
device(conf-ipacl-ext)# hard-drop tcp host 10::1 any count
device(conf-ipacl-ext)# hard-drop udp any host 20::1 count
device(conf-ipacl-ext)# permit tcp host 10::2 any eq telnet count
device(conf-ipacl-ext)# permit tcp host 10::2 any eq bgp count
device(conf-ipacl-ext)# hard-drop tcp host 10::3 host ff02::1 count

device(conf-ipacl-ext)# rb 1
device(config-rbridge-id-1)# ipv6 receive access-group ipv6-receive-acl-example in
```

## Related Commands

[ipv6 access-group](#), [ipv6 access-list](#), [show running-config ipv6 access-list](#)

# seq (IPv4 standard ACLs)

Inserts filtering rules in IPv4 standard ACLs. Standard ACLs permit or deny traffic according to source address only.

## Syntax

```
seq seq-value { deny | permit | hard-drop } { any | S_IPAddress mask | host S_IPAddress } [ count ] [ log ]
{ deny | permit | hard-drop } { any | S_IPAddress mask | host S_IPAddress } [ count ] [ log ]
no seq seq-value
no { deny | permit | hard-drop } { any | S_IPAddress mask | host S_IPAddress } [ count ] [ log ]
```

## Parameters

### seq

(Optional) Enables you to assign a sequence number to the rule. If you do not specify **seq seq-value**, the rule is added at the end of the list.

*seq-value*

Valid values range from 0 through 4294967290.

### permit

Specifies rules to permit traffic.

### deny

Specifies rules to deny traffic.

### hard-drop

Overrides the trap behavior for control frames and data frames such as echo request (ping). However, **hard-drop** does not override a **permit** for this address in a preceding rule.

### any

Specifies all source addresses.

### *S\_IPAddress*

Specifies a source address for which you want to filter the subnet.

*mask*

Defines a mask, whose effect is to specify a subnet that includes the source address that you specified. For options to specify the mask, see the Usage Guidelines.

### host

Specifies a source address.

*S\_IPAddress*

The source address.

### count

Enables statistics for the rule.

### log

(Available for **permit** or **deny** only) Enables logging for the rule. In addition, the ACL log buffer must be enabled, using the **debug access-list-log buffer** command.

## Modes

ACL configuration mode

## Usage Guidelines

This command configures rules to permit or drop traffic based on source addresses. You can also enable counters and logging for specified rules.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list.

The hard-drop option can prevent trapping of control frames. As a result, it could interfere with normal operations of the protocols.

For an ACL applied to a management interface, **hard-drop** keywords are interpreted as **deny** keywords.

You can specify a mask in either of the following ways:

- Wildcard mask format—for example, 0.0.0.255. The advantage of this format is that it enables you mask any bit, for example by specifying 0.255.0.255.
- Classless Interdomain Routing (CIDR) format—in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 as wildcard mask format.

If you are defining rules for a QoS ACL, be aware of the following considerations for ACLs implemented under flow-based QoS. For more information, refer to the "Configuring QoS" > "Flow-based QoS" section of the *Network OS Layer 2 Switching Configuration Guide*.

- Do not include the **count** keyword in ACLs intended for flow-based QoS implementation, because such ACLs automatically share a common counter.
- The **deny** keyword functions as a PASS THROUGH: For a match, QoS action defined for that class is not implemented.
- The **hard-drop** keyword is equivalent to the **deny** keyword.

To delete a rule from an ACL, do the relevant of the following:

- If you know the rule number, enter **no seq seq-value**.
- If you do not know the rule number, type **no** and then enter the full syntax without *seq-value*.

## Examples

The following example shows how to create a IPv4 standard ACL, define rules for it, and apply the ACL to an interface:

```
device# configure
device(config)# ip access-list standard stdACL3
device(config-ipacl-std)# seq 5 permit host 10.20.33.4
device(config-ipacl-std)# seq 15 deny any
device(config-ipacl-std)# exit
device(config)# interface ten 122/5/22
device(conf-if-te-122/5/22)# ipv4 access-group stdACL3 in
```

## Related Commands

[ip access-group](#), [ip access-list](#), [show running-config ip access-list](#)

# seq (IPv6 standard ACLs)

Inserts filtering rules in IPv6 standard ACLs. Standard ACLs permit or deny traffic according to source address only.

## Syntax

```
seq seq-value { deny | permit | hard-drop } { any | A:B:C:D:E:F:H:I / prefix_len | host S_IPAddress } [ count ] [ log ]
{ deny | permit | hard-drop } { any | A:B:C:D:E:F:H:I / prefix_len | host SIP_address | SIP_addressmask } [ count ] [ log ]
no seq seq-value
no { deny | permit | hard-drop } { any | A:B:C:D:E:F:H:I / prefix_len | host SIP_address | SIP_addressmask } [ count ] [ log ]
```

## Parameters

### seq

(Optional) Enables you to assign a sequence number to the rule. If you do not specify **seq seq-value**, the rule is added at the end of the list.

*seq-value*

Valid values range from 0 through 4294967290.

### permit

Specifies rules to permit traffic.

### deny

Specifies rules to deny traffic.

### hard-drop

Overrides the trap behavior for control frames and data frames such as echo request (ping). However, **hard-drop** does not override a **permit** for this address in a preceding rule.

### any

Specifies all source addresses.

### *S\_IPAddress*

Specify a source address for which you want to filter the subnet. For options to abbreviate the address, see the Usage Guidelines.

*prefix\_len*

Indicates how many of the high-order, contiguous bits of the address comprise the IPv6 prefix.

### host

Specifies a source address.

*SIP\_address*

The source address. For options to abbreviate the address, see the Usage Guidelines.

### count

Enables statistics for the rule.

### log

(Available for **permit** or **deny** only) Enables logging for the rule. In addition, the ACL log buffer must be enabled, using the **debug access-list-log buffer** command.

## Modes

ACL configuration mode

## Usage Guidelines

This command configures rules to permit or drop traffic based on source addresses. You can also enable counters and logging for specified rules.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list.

The hard-drop option can prevent trapping of control frames. As a result, it could interfere with normal operations of the protocols.

For an ACL applied to a management interface, **hard-drop** keywords are interpreted as **deny** keywords.

You can abbreviate an IPv6 address by using one or more of the following rules:

- Remove one or more leading zeros from one or more groups of hexadecimal digits; this is usually done to either all or none of the leading zeros. (For example, convert the group 0042 to 42.)
- Omit consecutive sections of zeros, using a double colon (::) to denote the omitted sections. The double colon may only be used once in any given address, as the address would be indeterminate if the double colon were used multiple times. A double colon may not be used to denote an omitted single section of zeros. (For example, 2001:db8::1:2 is valid, but 2001:db8::1::2 or 2001:db8::1:1:1:1:1 are not permitted.)

If you are defining rules for a QoS ACL, be aware of the following considerations for ACLs implemented under flow-based QoS. For more information, refer to the "Configuring QoS" > "Flow-based QoS" section of the *Network OS Layer 2 Switching Configuration Guide*.

- Do not include the **count** keyword in ACLs intended for flow-based QoS implementation, because such ACLs automatically share a common counter.
- The **deny** keyword functions as a PASS THROUGH: For a match, QoS action defined for that class is not implemented.
- The **hard-drop** keyword is equivalent to the **deny** keyword.

To delete a rule from an ACL, do the relevant of the following:

- If you know the rule number, enter **no seq seq-value**.
- If you do not know the rule number, type **no** and then enter the full syntax without *seq-value*.

## Examples

The following example shows how to create an IPv6 standard ACL and define rules for it:

```
device# configure terminal
device(config)# ipv6 access-list standard ipv6-std-acl
device(conf-ip6acl-std)# seq 10 permit host 0:1::1
device(conf-ip6acl-std)# seq 20 deny 0:2::/64
device(conf-ip6acl-std)# seq 30 hard-drop any count
```

## Related Commands

[ipv6 access-group](#), [ipv6 access-list](#), [show running-config ipv6 access-list](#)

## seq (MAC extended ACLs)

Inserts filtering rules in a Layer 2 (MAC) extended ACLs. Extended ACLs permit or deny traffic according to source and destination addresses, as well as other parameters.

### Syntax

```
seq seq-value { deny | permit | hard-drop } { any | SMAC_address mask | host SMAC_address } { any | host DMAC_address |
  DMAC_address mask } [ EtherType | arp | fcoe | ipv4 ] [ vlan vlanID ] [ count ] [ log ]
{ deny | permit | hard-drop } { any | SMAC_address mask | host SMAC_address } { any | host DMAC_address |
  DMAC_address mask } [ EtherType | arp | fcoe | ipv4 ] [ vlan vlanID ] [ count ] [ log ]
no seq seq-value
no seq { deny | permit | hard-drop } { any | SMAC_address mask | host SMAC_address } { any | host DMAC_address |
  DMAC_address mask } [ EtherType | arp | fcoe | ipv4 ] [ vlan vlanID ] [ count ] [ log ]
```

### Parameters

#### seq

(Optional) Enables you to assign a sequence number to the rule. If you do not specify **seq seq-value**, the rule is added at the end of the list.

*seq-value*

Valid values range from 0 through 4294967290.

#### permit

Specifies rules to permit traffic.

#### deny

Specifies rules to deny traffic.

#### hard-drop

Overrides the trap behavior for control frames and data frames such as echo request (ping). However, **hard-drop** does not override a **permit** for this address in a preceding rule.

#### any

Specifies all source MAC addresses.

#### SMAC\_address

Specifies a source MAC address and a comparison mask.

*mask*

Specify the mask using F's and zeros. For example, to match on the first two bytes of the address aabb.ccdd.eeff, use the mask ffff.0000.0000. In this case, the clause matches all MAC addresses that contain "aabb" as the first two bytes and any values in the remaining bytes.

#### host

Specifies a source MAC address.

*SMAC\_address*

Use the format HHHH.HHHH.HHHH.



**any**

Specifies all destination MAC addresses.

*DMAC\_address*

Specify a destination MAC address and a comparison mask.

*mask*

Specify the mask using F's and zeros. For example, to match on the first two bytes of the address aabb.ccdd.eeff, use the mask ffff.0000.0000. In this case, the clause matches all MAC addresses that contain "aabb" as the first two bytes and any values in the remaining bytes.

**host**

Specifies a destination MAC address.

*DMAC\_address*

Use the format HHHH.HHHH.HHHH.

*EtherType*

Specifies the protocol number for which to set the permit or deny conditions. Valid values range from 1536 through 65535.

**arp**

Specifies to permit or deny the Address Resolution Protocol (0x0806).

**fcoe**

Specifies to permit or deny the Fibre Channel over Ethernet Protocol (0x8906).

**ipv4**

Specifies to permit or deny the IPv4 protocol (0x0800).

vlan *vlanID*

Specifies a VLAN interface to which the ACL is bound.

**count**

Enables statistics for the rule.

**log**

(Available for **permit** or **deny** only) Enables logging for the rule. In addition, the ACL log buffer must be enabled, using the **debug access-list-log buffer** command.

## Modes

ACL configuration mode

## Usage Guidelines

This command configures rules to permit or drop traffic based on source and destination MAC addresses and protocol type. You can also enable counters and logging for specific rules.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list.

The hard-drop option can prevent trapping of control frames. As a result, it could interfere with normal operations of the protocols.

If you are defining rules for a QoS ACL, be aware of the following considerations for ACLs implemented under flow-based QoS. For more information, refer to the "Configuring QoS" > "Flow-based QoS" section of the *Network OS Layer 2 Switching Configuration Guide*.

- Do not include the **count** keyword in ACLs intended for flow-based QoS implementation, because such ACLs automatically share a common counter.
- The **deny** keyword functions as a PASS THROUGH: For a match, QoS action defined for that class is not implemented.
- The **hard-drop** keyword is equivalent to the **deny** keyword.

To delete a rule from an ACL, do the relevant of the following:

- If you know the rule number, enter **no seq seq-value**.
- If you do not know the rule number, type **no** and then enter the full syntax without *seq-value* .

## Examples

The following example creates a rule in a MAC extended ACL to deny IPv4 traffic from the source MAC address 0022.3333.4444 to the destination MAC address 0022.3333.5555 and to enable the counting of packets.

```
device# configure terminal
device(config)# mac access-list extended ACL1
device(conf-macl-ext)# seq 100 deny 0022.3333.4444 0022.3333.5555 ipv4 count
```

The following example deletes a rule from a MAC extended ACL.

```
device# configure terminal
device(config)# mac access-list extended ACL1
device(conf-macl-ext)# no seq 100
```

# seq (MAC standard ACLs)

Inserts filtering rules in Layer 2 (MAC) standard ACLs. Standard ACLs permit or deny traffic according to source address only.

## Syntax

```
seq seq-value { deny | permit | hard-drop } { any | SMAC_address mask | host SMAC_address } [count ] [ log ]
{ deny | permit | hard-drop } { any | SMAC_address mask | host SMAC_address } [count ] [ log ]
no seq seq-value
no seq { deny | permit | hard-drop } { any | SMAC_address mask | host SMAC_address } [count ] [ log ]
```

## Parameters

### seq

(Optional) Enables you to assign a sequence number to the rule. If you do not specify **seq seq-value**, the rule is added at the end of the list.

*seq-value*

Valid values range from 0 through 4294967290.

### permit

Specifies rules to permit traffic.

### deny

Specifies rules to deny traffic.

### hard-drop

Overrides the trap behavior for control frames and data frames such as echo request (ping). However, **hard-drop** does not override a **permit** for this address in a preceding rule.

### any

Specifies all source MAC addresses.

### SMAC\_address

Specifies a source MAC address and a comparison mask.

*mask*

Specify the mask using F's and zeros. For example, to match on the first two bytes of the address aabb.ccdd.eeff, use the mask ffff.0000.0000. In this case, the clause matches all MAC addresses that contain "aabb" as the first two bytes and any values in the remaining bytes.

### host

Specifies a source MAC address.

*SMAC\_address*

Use the format HHHH.HHHH.HHHH.

### count

Enables statistics for the rule.

### log

(Available for **permit** or **deny** only) Enables logging for the rule. In addition, the ACL log buffer must be enabled, using the **debug access-list-log buffer** command.

## Modes

ACL configuration mode

## Usage Guidelines

This command configures rules to permit or drop traffic based on source MAC address. You can also enable counters and logging for specific rules.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list.

The hard-drop option can prevent trapping of control frames. As a result, it could interfere with normal operations of the protocols.

If you are defining rules for a QoS ACL, be aware of the following considerations for ACLs implemented under flow-based QoS. For more information, refer to the "Configuring QoS" > "Flow-based QoS" section of the *Network OS Layer 2 Switching Configuration Guide*.

- Do not include the **count** keyword in ACLs intended for flow-based QoS implementation, because such ACLs automatically share a common counter.
- The **deny** keyword functions as a PASS THROUGH: For a match, QoS action defined for that class is not implemented.
- The **hard-drop** keyword is equivalent to the **deny** keyword.

To delete a rule from an ACL, do the relevant of the following:

- If you know the rule number, enter **no seq seq-value**.
- If you do not know the rule number, type **no** and then enter the full syntax, without **seq seq-value**.

## Examples

The following command creates statistic-enabled rules in a MAC standard ACL:

```
device# configure terminal
device(config)# mac access-list standard ACL1
device(conf-macl-std)# seq 100 deny host 0022.3333.4444 count
device(conf-macl-std)# seq 110 permit host 0011.3333.5555 count
```

The following command deletes a rule in a MAC standard ACL:

```
device# configure terminal
device(config)# mac access-list standard ACL1
device(conf-macl-std)# no seq 100
```

## Related Commands

[mac access-group](#), [mac access-list standard](#), [resequence access-list](#), [show running-config mac access-list](#)

# service password-encryption

Enables a global password encryption policy that overrides **username** encryption settings.

## Syntax

**service password-encryption**

**no service password-encryption**

## Command Default

Global password encryption policy is enabled.

## Modes

Global configuration mode

## Usage Guidelines

If global password encryption policy is enabled, it overrides **username** encryption settings.

To disable global password encryption policy, enter the **no** form of this command.

Even if global password encryption policy is disabled, the following **username** syntax does encrypt that user's password: **encryption-level 7**.

## Examples

The following example enables global password encryption policy.

```
device# configure terminal
device(config)# service password-encryption
```

The following example disables global password encryption policy.

```
device# configure terminal
device(config)# no service password-encryption
```

## Related Commands

[show running-config password-attributes](#)

# service-policy

Binds a policy map to an interface.

## Syntax

**service-policy** in | out *policy-mapname*

**no service-policy** in | out

## Command Default

No service policy is created.

## Parameters

**in**

Binds policy map to inbound traffic.

**out**

Binds policy map to outbound traffic.

*policy-mapname*

Name of the policy map.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command applies a policy-map containing a class-map with specific Policer parameters and match critters to a switch interface. The policy map must be configured before you can apply it (refer to the description of the **policy-map** command).

The **no** form of this command removes the service policy.

### NOTE

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

## Examples

To create a service policy for outbound traffic on a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 237/1/8
device(conf-if-te-237/1/8)# service-policy out policymap1
```

To remove a service policy for outbound traffic from a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 237/1/8
device(conf-if-te-237/1/8)# no service-policy out
```

To remove a service-policy for inbound traffic on a specific 10-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 237/1/8
device(conf-if-te-237/1/8)# no service-policy in
```

# set as-path

Sets a prepended string or a tag for a BGP AS-path attribute in a route-map instance.

## Syntax

```
set as-path [ prepend string | tag ]
```

```
no set as-path [ prepend string | tag ]
```

## Parameters

### prepend

Prepends the string to the AS-path.

*string*

AS numbers. Range is from 1 through 4294967295.

### tag

Sets a route tag.

## Modes

Route-map configuration mode

## Examples

Typical command execution:

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# routemap myroutemap1 permit 10
device(config-route-map-myroutemap1/permit/10)# set as-path prepend 7701000
```

## Related Commands

[route-map](#)



# set as-path prepend

Prepends an AS4 number to an AS path, makes the AS number a tag attribute for a route map, and provides a variety of route-management options.

## Syntax

```
set as-path prepend as-num , as-num , . . . as-num [ automatic-tag ] [ [ comm-list acl delete ] ] [ [ community num : num | num | additive | internet | local-as | no-advertise | no-export ] ] [ [ dampening [ half-life | reuse | suppress | max-suppress-time ] ] ] [ [ ip next hop ip-addr ] [ ip next-hop peer-address ] ] [ [ local-preference num ] ] [ [ metric [ add | assign | none | sub ] ] ] [ [ metric-type [ type-1 | type-2 ] ] ] [ [ external [ metric-type internal ] ] ] [ [ origin igp | incomplete ] ] [ [ tag ] ] [ [ weight num ] ]
```

```
no set as-path prepend as-num , as-num , . . . as-num
```

## Parameters

### *automatic-tag*

Calculates and sets an automatic tag for the route.

### **comm-list** *acl delete*

Deletes a community from the community attributes field for a BGP4 route.

### **community**

Sets the community attribute for the route to the number or well-known type specified. Possible values are *num* : *num* , Internet, no-export, local-as, no-advertise.

#### *num:num*

Specific community member.

### **additive**

Adds a community to the already existing communities.

### **internet**

The Internet community.

### **local-as**

Local sub-AS within the confederation. Routes with this community can be advertised only within the local sub-AS.

### **no-advertise**

Routes with this community cannot be advertised to any other BGP4 devices at all.

### **no-export**

Community of sub-ASs within a confederation. Routes with this community can be exported to other sub-ASs in the same confederation but not outside the confederation to other ASs or otherwise sent to EBGP neighbors.

### **dampening**

Sets dampening parameters for the route.

#### *half-life*

Number of minutes after which the route penalty becomes half its value.

*reuse*

Specifies how low a route penalty must become before the route becomes eligible for use again after being suppressed.

*suppress*

Specifies how high a route penalty can become before the device suppresses the route.

*max-suppress-time*

Specifies the maximum number of minutes that a route can be suppressed regardless of how unstable it is.

**ip next hop**

Sets the next-hop IP address for a route that matches the match statement in the route map.

*ip-addr*

IPv4 address in dotted-decimal notation.

**ip next-hop peer-address**

Sets the BGP4 next hop for a route to the neighbor address.

**local-preference**

Sets the local preference for the route.

*num*

Range is from 0 through 4294967295.

**metric**

Sets the MED (metric) value for the route. Range is from 0 through 4294967295. The default MED value is 0.

**add**

Adds to the current metric value.

**assign**

Replaces the current metric value with a new value.

**none**

Removes the MED attribute (metric) from the BGP4 route.

**sub**

Subtracts from the current metric value.

**metric-type**

Changes the metric type of the route redistributed into OSPF.

**type-1**

Type 1 route.

**type-2**

Type 2 route.

**external**

External Type 1 or Type 2 route.

**metric-type internal**

Sets route MED attribute to same value as the IGP metric of the BGP4 next-hop route, for advertising a BGP4 route to an EBGp neighbor.

**next-hop**

Sets IPv4 address of the next hop.

*ip-addr*  
IPv4 address in dotted-decimal notation.

**origin**

Sets the route's origin.

**igp**

Sets origin to IGP.

**incomplete**

Sets origin to INCOMPLETE.

**tag**

Keyword that makes the ASN an AS-path tag attribute. (Applies only to routes redistributed into OSPF.)

**weight**

Sets the weight for the route.

*num*

Range is from 0 through 4294967295.

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the configuration.

## Examples

To prepend an AS4 number:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set as-path prepend 7701000
```

## Related Commands

[route-map](#)

# set automatic-tag

Sets the route-map tag value.

## Syntax

`set automatic-tag value`

## Parameters

*value*

The value for the computed tag.

## Modes

Route-map configuration mode

## Usage Guidelines

This command sets an automatically computed tag value in a route-map instance.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set automatic-tag 5
```

## Related Commands

[route-map](#)

# set comm-list

Sets a BGP community list for deletion in a route-map instance.

## Syntax

**set comm-list** *name*

**no set comm-list** *name*

## Parameters

*name*

BGP community list name. Range is from 1 through 32 ASCII characters.

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** version of this command to disable this feature.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set comm-list test
```

## Related Commands

[route-map](#)

# set community

Sets a BGP community attribute in a route-map instance.

## Syntax

**set community** [*community-number* | additive | internet | local-as | no-advertise | no-export | none]

**no set community** *community-number*

## Parameters

*community-number*

BGP community number, in two format options:(1) Range is from 1 through 4294967295.(2) Format is AA:NN, where AA is the AS number, and NN is a locally significant number.

**additive**

Add to the existing community.

**internet**

Send to internet (well-known community).

**local-as**

Do not send outside local AS (well-known community).

**no-advertise**

Do not advertise to any peer (well-known community).

**no-export**

Do not export to next AS (well-known community).

**none**

Sets no community attribute.

## Modes

Route-map configuration mode

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set community no-export
```

## Related Commands

[route-map](#)

# set cos traffic-class

Specifies the User-Priority field value in VLAN header and traffic-class queuing value when a packet matches a flow.

## Syntax

```
set cos { 0..7 } traffic-class { 0..7 }  
no set cos { 0..7 } traffic-class { 0..7 }
```

## Parameters

0..7

Modifies the Class of Service (CoS) value in the VLAN header of classified traffic, or assigns a queue to the classified traffic. The range of valid values is from 0 through 7.

## Modes

Route-map configuration mode

## Examples

Typical command example:

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# route-map myroutes  
device(config-route-map myroutes)# set cos traffic-class 1
```

## Related Commands

[class](#), [policy-map](#)

# set dampening

Sets a BGP route-flap dampening penalty in a route-map instance.

## Syntax

**set dampening** *number*

**no set dampening** *number*

## Command Default

The default is 15.

## Parameters

*number*

Half-life in minutes for the penalty. Range is from 1 through 45.

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of this commands removes the penalty.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set dampening 25
```

## Related Commands

[route-map](#)



# set distance

Sets the administrative distance for matching OSPF routes in route-map instance.

## Syntax

**set distance** *value*

**no set distance**

## Parameters

*value*

Administrative distance for the route. Range is from 1 through 254.

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of this command removes the configuration.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set distance 50
```

## set dscp

Specifies the DSCP field value in IP header when a packet matches a flow.

### Syntax

```
set dscp { 0..63 }  
no set dscp { 0..63 }
```

### Parameters

0..63

The DSCP value in the IP header of the classified traffic. The range of valid values is from 0 through 63.

### Modes

Route-map configuration mode

### Examples

Typical command example:

```
device# configure terminal  
device(config)# rbridge-id 10  
device(config-rbridge-id-10)# route-map myroutes  
device(config-route-map myroutes)# set dscp 50
```

### Related Commands

[class](#), [policy-map](#)

# set extcommunity

Sets an extended BGP community attribute in a route-map instance.

## Syntax

```
set extcommunity { rt extcommunity value | soo extcommunity value }
no set extcommunity
```

## Command Default

No extended BGP community attribute is set.

## Parameters

**rt**  
Specifies the route target (RT) extended community attribute.

**soo**  
Specifies the site of origin (SOO) extended community attribute.

*extcommunity value*  
Specifies the value. The value can be one of the following:

- ASN:nn—autonomous-system-number:network-number  
Autonomous system (AS) number and network number.
- IPAddress:nn—ip-address:network-number  
IP address and network number.

## Modes

Route-map configuration mode.

## Usage Guidelines

Enter **no set extcommunity** to delete an extended community set statement from the configuration file.

## Examples

The following example sets the route target to extended community attribute 1:1 for routes that are permitted by the route map.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# route-map extComRmap permit 10
device(config-route-map-sendExtComRmap/permit/10)# set extcommunity rt 1:1
```

set extcommunity

The following example sets the site of origin to extended community attribute 2:2 for routes that are permitted by the route map.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ip community-list extended 1 permit 123:2
device(config-rbridge-id-122)# route-map extComRmap permit 10
device(config-route-map-sendExtComRmap/permit/10)# set extcommunity soo 2:2
```

## History

Release version	Command history
5.0.0	This command was introduced.

# set ip interface null0

Drops traffic when the null 0 statement becomes the active setting as determined by the route-hop selection process for IPv4 policy-based routing.

## Syntax

```
set ip interface null0
no set ip interface null0
```

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of this command deletes the matching filter from the ACL.

## Examples

The following example configures the next hop as NULL0 interface to send the traffic to the null interface, thus dropping the packet instead of forwarding it.:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set ip interface null0
```

# set ip next-hop

Sets the IPv4 address of the next hop in a route-map instance.

## Syntax

```
set ip [ global | vrf vrf-name ] next-hop A.B.C.D
```

```
no set ip next-hop A.B.C.D
```

## Parameters

*A.B.C.D*

IPv4 address of the next hop.

**global**

Specifies that the next specified hop address is to be resolved from the global routing table.

**vrf** *vrf-name*

Specifies from which VRF routing table the specified next hop address will be resolved.

**next hop** *A.B.C.D*

Sets the next hop to which to route the packet. The next hop must be adjacent.

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to delete the matching filter from the ACL.

When a route-map is applied to BGP, and the route-map has multiple **set ip next-hop** statements in a single instance, BGP considers the last **set ip next-hop** in the route-map.

## Examples

The following example specifies that the next specified hop address is to be resolved from the global routing table.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set ip next-hop global
```

# set ipv6 next-hop

Sets the IPv6 address of the next hop in a route-map instance.

## Syntax

```
set ipv6 [ global | vrf vrf-name ] next-hop AAAA:BBBB:CCCC:DDDD:EEEE:FFFF:GGGG:HHHH
no set ipv6 next-hop AAAA:BBBB:CCCC:DDDD:EEEE:FFFF:GGGG:HHHH
```

## Parameters

*AAAA:BBBB:CCCC:DDDD:EEEE:FFFF:GGGG:HHHH*

IPv6 address of the next hop.

**global**

Specifies that the next specified hop address is to be resolved from the global routing table.

**vrf** *vrf-name*

Specifies from which VRF routing table the specified next hop address will be resolved.

**next hop** *AAAA:BBBB:CCCC:DDDD:EEEE:FFFF:GGGG:HHHH*

Sets the next hop to which to route the packet. The next hop must be adjacent.

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to delete the matching filter from the ACL.

## Examples

The following example specifies that the next specified hop address is to be resolved from the global routing table.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set ipv6 next-hop global
```

# set local-preference

Sets a BGP local-preference path attribute in a route-map instance.

## Syntax

**set local-preference** *number*  
**no set local-preference**

## Parameters

*number*  
Range is from 0 through 4294967295.

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the attribute.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set local-preference 8675309
```

## Related Commands

[route-map](#)



# set metric

Configures the route metric set clause in a route-map instance.

## Syntax

```
set metric [ add | assign | sub ] value
```

```
no set metric [ add | assign | sub ] value
```

## Parameters

### add

Adds the value to the current route metric.

### assign

Replaces the current route metric with this value.

### sub

Subtracts the value from the current route metric.

### none

Removes the current route metric.

*value*

Range is from 0 through 4294967295.

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the configuration.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set metric add
```

# set metric-type

Sets a variety of metric types for destination routing in a route-map instance.

## Syntax

```
set metric-type [ external | internal | type-1 | type-2 ]
```

```
no set metric-type [ external | internal | type-1 | type-2 ]
```

## Parameters

*external*

IS-IS external metric

**internal**

IGP internal metric to BGP MED

**type-1**

OSPF external type-1 metric

**type-2**

OSPF external type-2 metric

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the configuration.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set metric-type internal
```

## Related Commands

[route-map](#)

# set origin

Sets a BGP origin code in a route-map instance.

## Syntax

```
set origin [ igp | incomplete ]
```

```
no set origin [ igp | incomplete ]
```

## Parameters

*igp*

Local IGP

**incomplete**

Unknown heritage

## Modes

Route-map configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the configuration.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set origin incomplete
```

## Related Commands

[route-map](#)

# set-priority

Configures the priority map of a class map.

## Syntax

**set-priority** *priority-map-name*

**no set-priority** *priority-map-name*

## Parameters

*priority-map-name*

The priority-map name that you are including in the policy map. Refer to the description of the **police-priority-map** command.

## Modes

Policy map - class map configuration mode

## Usage Guidelines

Only the **police cir** and **cbs** commands are mandatory for configuring a class map.

If the optional parameters for a class map are not set, they are treated as disabled. To delete parameters for a class map, you must delete all policer parameters while in the policy map class configuration mode using the **no police cir** command.

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later switches.

Use the **no** form of this command to remove the parameter from the class map.

## Examples

```
device# configure terminal
device(config)# mac access-list standard acl1
device(conf-macl-std)# permit any
device(conf-macl-std)# class-map class1
device(config-classmap)# match access-group acl1
device(config-classmap)# policy-map policy1
device(config-policymap)# class default
device (config-policymap-class)# police cir 40000
device (config-policymap-class)# set-priority default
```

# set route-type

Sets a route type in a route-map instance.

## Syntax

```
set route-type [ internal | type-1 | type-2 ]
```

```
no set route-type
```

## Parameters

*internal*

Internal route type

**type-1**

OSPF external route type 1

**type-2**

OSPF external route type 2

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of this command removes the configuration.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set route-type type-1
```

# set tag

Sets the route tag value in a route-map instance.

## Syntax

**set tag** *value*

**no set tag** *value*

## Parameters

*value*

The tag clause value for the route-map. Range is from 0 through 4294967295.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **no** form of this command disables this feature.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set tag 8675309
```

# set weight

Sets a BGP weight for the routing table in a route-map instance.

## Syntax

**set weight** *number*

**no set weight** *number*

## Parameters

*number*

Weight value. Range is from 0 through 65535.

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of this command disables this feature

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map myroutes
device(config-route-map myroutes)# set weight 500
```

## Related Commands

[route-map](#)

# sflow (VXLAN)

Enables sFlow monitoring of the tunnel endpoints for a VXLAN overlay gateway site.

## Syntax

```
sflow profile_name remote-endpoint { IPv4_address | any } vlan { add | remove } vlan_id [ vrf name ]
no sflow profile_name
```

## Parameters

*profile\_name*

Name of a configured sFlow profile.

**remote-endpoint**

Specifies an IPv4 address or all IPv4 addresses associated with the remote tunnel endpoint for the site.

*IPv4\_address*

IPv4 address for the tunnel endpoint.

**any**

Specifies all IPv4 addresses for the tunnel endpoint.

**vlan**

Specifies a VLAN ID or range of VLAN IDs to be added or removed from the tunnel.

**add**

Specifies a VLAN ID or range of VLAN IDs to be added to the tunnel.

**remove**

Specifies a VLAN ID or range of VLAN IDs to be removed from the tunnel.

*vlan\_id*

A VLAN ID or range of VLAN IDs. See the Usage Guidelines.

**vrf**

Specifies a VRF instance.

*name*

Name of the VRF instance.

## Modes

VXLAN overlay gateway site configuration mode

## Usage Guidelines

An sFlow profile must be configured, by means of the **sflow-profile** command.

Use the **no sflow profile\_name** command to remove the configuration from the overlay gateway.



## Examples

To enable sFlow monitoring for all endpoints for specified VLAN IDs:

```
device# configure terminal
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# sflow my_sflow_profile remote-endpoint any vlan add 10,20-30
```

# sflow collector

Configures the forwarding of sFlow datagrams to collectors.

## Syntax

```
sflow collector { IPv4address | IPv6address } { port_num } [ use-vrf vrf-name ]
no sflow collector { IPv4address | IPv6address } [ port_num ] [ use-vrf vrf-name ]
```

## Parameters

*IPv4address*

Specifies an IPv4 address in dotted-decimal format for the collector.

*IPv6address*

Specifies an IPv6 address for the collector.

*port\_num*

Specifies the port number to use for sending data to the collector. Range is 1 through 65535. The default is 6343.

**use-vrf** *vrf-name*

Specifies a VRF through which to connect to the collector. See the Usage Guidelines.

## Modes

Global configuration mode

## Usage Guidelines

You can only specify up to five sFlow collectors; this includes all VRFs.

Use the **no** form of this command to reset the specified collector address to a null value.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

To specify the sFlow collectors for an IPv4 address with the default port on the management VRF:

```
device# configure terminal
device(config)# sflow collector 192.10.138.176
```

To specify the sFlow collectors for an IPv4 address with a nondefault port on a user-specified VRF:

```
device# configure terminal
device(config)# sflow collector 192.10.138.176 50 use-vrf myvrf
```

To specify the sFlow collectors for an IPv6 address with a nondefault port on the management VRF:

```
device# configure terminal
device(config)# sflow collector 3ff3:1900:4545:3:200:f8ff:fe21:67cf:6343 50
```

## History

Release version	Command history
6.0.1	This command was modified to support the <b>use-vrf</b> keyword.
7.0.0	This command was modified to support user-specified VRF names.

# sflow enable (global version)

Enables sFlow globally.

## Syntax

**sflow enable**

**no sflow enable**

## Command Default

sFlow is disabled on the system.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported on physical ports only.

On an Extreme VDX 8770, SPAN and sFlow can be enabled at the same time.

The **no** form of this command disable sFlow globally.

## Examples

To enable sFlow globally:

```
device# configure terminal
device(config)# sflow enable
```

## History

Release version	Command history
17r.1.01	This command was introduced..

## Related Commands

[sflow enable \(interface version\)](#)

# sflow enable (interface version)

Enables sFlow on an interface. sFlow is used for monitoring network activity.

## Syntax

**sflow enable**

**no sflow enable**

## Command Default

sFlow is disabled on all interfaces.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command is supported on physical ports only.

On an Extreme VDX 8770 device, SPAN and sFlow can be enabled at the same time.

The **no** form of this command disable sFlow on an interface.

## Examples

To enable sFlow on a specific 40-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# sflow enable
```

## Related Commands

[interface](#), [sflow enable \(global version\)](#), [sflow polling-interval \(interface version\)](#), [sflow sample-rate \(interface version\)](#)

# sflow polling-interval (global version)

Configures the polling interval globally.

## Syntax

**sflow polling-interval** *interval\_value*

**no sflow polling-interval**

## Parameters

*interval\_value*

Specifies a value in seconds to set the polling interval. Valid values range from 1 through 65535 seconds.

## Command Default

The default is 20.

## Modes

Global configuration mode

## Usage Guidelines

The interval is the maximum number of seconds between successive samples of counters to be sent to the collector.

The **no** form of this command restores the default value.

## Examples

To set the polling interval to 135 seconds:

```
device# configure terminal
device(config)# sflow polling-interval 135
```

## Related Commands

[sflow polling-interval \(interface version\)](#)

# sflow polling-interval (interface version)

Configures the polling interval at the interface level.

## Syntax

**sflow polling-interval** *interval\_value*

**no sflow polling-interval**

## Command Default

The default is 20.

## Parameters

*interval\_value*

Specifies a value in seconds to set the polling interval. Valid values range from 1 through 65535.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The interval is the maximum number of seconds between successive samples of counters to be sent to the collector.

The **no** form of this command restores the default value.

## Examples

To set the polling interval to 135 seconds on a specific 40-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# sflow polling-interval 135
```

# sflow sample-rate (global version)

Sets the number of packets that are skipped before the next sample is taken.

## Syntax

**sflow sample-rate** *samplerate*

**no sflow sample-rate**

## Command Default

The default is 32768.

## Parameters

*samplerate*

Specifies the sampling rate value in packets. Valid values range from 2 through 16777215 packets.

## Modes

Global configuration mode

## Usage Guidelines

Sample-rate is the average number of packets skipped before the sample is taken.

The **no** form of this command restores the default sampling rate.

## Examples

To change the sampling rate to 4096:

```
device# configure terminal
device(config)# sflow sample-rate 4096
```

## Related Commands

[sflow polling-interval \(interface version\)](#)



# sflow sample-rate (interface version)

Sets the default sampling rate for an interface.

## Syntax

**sflow sample-rate** *samplerate*

**no sflow sample-rate**

## Command Default

The default is 32768.

## Parameters

*samplerate*

Specifies the sampling rate. Valid values range from 2 through 16777215 packets.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The default sampling rate determines how many packets are skipped before the next sample is taken for that interface. The default sampling rate of an interface is set to the same value as the current global default sampling rate.

This command changes the sampling rate for an interface.

The **no** form of this command restores the default setting.

## Examples

To change the sampling rate to 4096 packets on a specific 40-gigabit Ethernet interface:

```
device# configure terminal
device(config)# interface fortygigabitethernet 1/3/1
device(conf-if-fo-1/3/1)# sflow sample-rate 4096
```

# sflow source-ip

Specifies the IPv4 or IPv6 address of either the chassis (virtual IP address) or the local Management Module as the source of sFlow packets.

## Syntax

```
sflow source-ip { chassis-ip | mm-ip }
```

```
no sflow source-ip
```

## Command Default

sFlow uses the chassis virtual IP address by default.

## Parameters

### chassis-ip

Specifies the virtual IPv4 or IPv6 address of the chassis as the source of sFlow packets.

### mm-ip

Specifies the IPv4 or IPv6 address of the local Management Module as the source of sFlow packets.

## Modes

Global configuration mode

## Usage Guidelines

The "no" form of the command is available once the source type has been specified.

The chassis virtual IP address is configured by means of the **chassis** command.

The IP address of the local Management Module is configured by means of the **interface management** command.

## Examples

To specify the virtual IP address of the chassis as the source of sFlow packets:

```
device# config
device(config)# sflow source-ip chassis-ip
```

To specify the IP address of the local Management Module as the source of sFlow packets:

```
device(config)# sflow source-ip mm-ip
```

To confirm the above configuration:

```
device(config)# do show running-config sflow
sflow enable
sflow source-ip mm-ip
```

To disable the above configuration and revert to the default:

```
device(config)# no sflow source-ip
device(config)# do show running-config sflow
sflow enable
```

## History

Release version	Command history
6.0.1a	This command was introduced.

# sflow-profile

Establishes an sFlow profile name and sets a sampling rate.

## Syntax

```
sflow-profile { sflow_profile_name } { sample-rate sampling_rate }  
no sflow-profile { sflow_profile_name }
```

## Command Default

This command is disabled.

## Parameters

*sflow\_profile\_name*

Name of an sFlow profile for sampling rates. The maximum number of characters is 64.

**sample-rate**

Selects a sampling rate.

*sampling\_rate*

Specifies a sampling rate. Range is from 2 through 8388608 packets, in powers of 2 only. The default is 32768 packets.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command disables the sFlow profile.

## Examples

To establish an sFlow profile and set a sampling rate of 4096:

```
device# configure terminal  
device(config)# sflow mysflowprofile sample-rate 4096
```

# sflow profile-map

Attaches an sFlow profile map to a class map in flow-based QoS. This configures the sampling rate and other sFlow attributes.

## Syntax

```
sflow profile-map map_name
```

## Command Default

The sFlow profile map must be created.

## Parameters

*map\_name*

The sFlow profile map to attach to the class map in flow-based QoS. The maximum number of characters is 64.

## Modes

Policy-map configuration mode

## Usage Guidelines

Specifying a non-existent map name causes an error.

This action is allowed only in the ingress direction.

It can be configured both in user-defined class maps and in the class map "default". If configured in the class map "default", port-based sFlow is enabled.

## Examples

Typical command execution.

```
device# configure terminal
device(config)# policy-map policymap1
device(config-policyclass)#class cmap1
device(config-policyclass)#sflow profile-map mysflowprofile
```

## History

Release version	Command history
5.0.0	This command was introduced.

# sfp breakout

Allows a single physical 40G port to be utilized as multiple 10G ports. For example, a 40G port can be configured to operate as four individual 10G external ports.

## Syntax

**sfp breakout**

**no sfp breakout**

## Command Default

Breakout mode is set to disabled.

## Modes

Connector configuration mode

## Usage Guidelines

If you do not specify a speed, the system automatically configures the port into multiple 10G ports.

The port must not be a member of a port channel.

### NOTE

For the 27x40 GbE line card on an Extreme VDX 8770, a port group must be in performance mode before you can configure one of its ports to breakout mode.

Use the **no sfp breakout** command to disable breakout mode and restore the interface. For VDX 6740, VDX 6740T, VDX 6740T-1G, VDX 6940-36Q, and VDX 6940-144S, a reboot is required after using the **no sfp breakout** command.

## Examples

The following example enables SFP breakout on a connector.

```
device# configure terminal
device(config)# hardware
device(config-hardware)# connector 2/0/1
device(config-connector-2/0/1)# sfp breakout
```

The following example disables SFP breakout on a connector.

```
device# configure terminal
device(config)# hardware
device(config-hardware)# connector 2/0/1
device(config-connector-2/0/1)# no sfp breakout
```

## Related Commands

[power-off linecard](#), [power-on linecard](#)

# shape

Specifies the shaping rate for a port to smooth out the traffic egressing an interface

## Syntax

**shape** *speed*

## Parameters

*speed*

The speed for the shape rate in Kbps. The range of valid values is from 28000 to the top speed on the interface.

## Modes

Policymap configuration mode

## Usage Guidelines

This command is allowed only for the Egress direction.

This command can only be configured in for the **class class-default** command.

This command is mutually exclusive of the **scheduler** and **police** commands.

The minimum speed for a VDX 6740 is 200,000 Kbps.

## Examples

Typical command example:

```
device# configure terminal
device(config)#policy-map mutation
device(config-policymap)#class class-default
device(config-policyclass)# shape 30000
```

## Related Commands

[class](#), [policy-map](#)

# short-path-forwarding

Enables short-path forwarding on a VRRP Extended (VRRP-E) router.

## Syntax

```
short-path-forwarding [ revert-priority number ]  
no short-path-forwarding
```

## Command Default

Backup routers do not forward packets.

## Parameters

**revert-priority** *number*

Allows additional control over short-path-forwarding on a backup router. If you configure this option, the revert-priority number acts as a threshold for the current priority of the session, and only if the current priority is higher than the revert-priority will the backup router be able to route frames. The range of revert-priority is 1 to 254.

## Modes

Virtual-router-extended-group configuration mode

## Usage Guidelines

Short-path forwarding allows a backup router within a virtual router to bypass the VRRP-E master router and directly forward packets through the interfaces on that same backup router.

This command can be used for VRRP-E, but not for VRRP. You can perform this configuration on a virtual Ethernet (VE) interface only.

Enter **no short-path-forwarding** to remove this configuration.

## Examples

To enable short-path-forwarding on a VRRP-E group:

```
device# configure terminal  
device(config)# rbridge-id 101  
device(config-rbridge-id-101)# interface ve 25  
device(config-ve-25)# vrrp-extended-group 100  
device(config-vrrp-extended-group-100)# short-path-forwarding
```

To enable short-path-forwarding with a revert-priority threshold of 95:



# Show commands

---

## show access-list

For a given network protocol and inbound/outbound direction, displays ACL status information. You can show information for a specified ACL or only for that ACL on a specified interface. You can also display information for all ACLs bound to a specified device interface, VLAN, VE, or VXLAN overlay-gateway. You can also display information for receive-path ACLs on a specified RBridge or for all RBridges.

### Syntax

```
show access-list { ip | ipv6 | mac }
show access-list { ip | ipv6 | mac } name { in | out }
show access-list interface { <N>gigabitethernet rbridge_id/slot/port | port-channel index | vlan vlan_id } { in | out }
show access-list interface management rbridge_id/port in
show access-list interface ve vlan_id { in | out } [ rbridge-id { rbridge_id | all } ]
show access-list overlay-gateway overlay_gateway_name in
show access-list mac name interface { <N>-gigabitethernet rbridge_id/slot/port | port-channel index | vlan vlan_id } { in | out }
show access-list { ip | ipv6 } name interface { <N>gigabitethernet rbridge_id/slot/port | port-channel index } { in | out }
show access-list { ip | ipv6 } name interface ve vlan_id in | out [ rbridge-id { rbridge_id | all } ]
show access-list { ip | ipv6 } name interface management rbridge_id/port in
show access-list { ip | ipv6 } name rbridge-id { rbridge_id | all } in
show access-list rbridge-id { rbridge_id | all } in
```

### Parameters

**ip | ipv6 | mac**

Specifies the network protocol.

*name*

Specifies the ACL name.

**interface**

Filters by interface.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge\_id*

Specifies an RBridge ID.

- slot*  
Specifies a valid slot number.
- port*  
Specifies a valid port number.
- management** *rbridge\_id/port*  
Specifies a management interface.
- port-channel** *index*  
Specifies a port-channel interface.
- vlan** *vlan\_id*  
Specifies a VLAN interface.
- ve** *vlan\_id*  
Specifies a virtual Ethernet (VE) interface.
- rbridge-id**  
Specifies one or all RBridges.
- rbridge\_id*  
Specifies an RBridge.
- all**  
Specifies all RBridges.
- overlay-gateway** *overlay\_gateway\_name*  
Specifies a VXLAN overlay-gateway.
- in | out**  
Specifies the ACL binding direction (incoming or outgoing).

## Modes

Privileged EXEC mode

## Usage Guidelines

On the Extreme VDX family of hardware, VLANs are treated as interfaces from a configuration point of view. By default all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). For details of valid VLAN IDs, refer to the **vlan classifier group** command.

## Command Output

The **show access-list** command displays the following information:

Output field	Description
Active	The rule is active and implements the configured action.
Partial	The rule is partially programmed, with the configured action implemented in some cases. This is typically seen for logical interfaces like VLAN, which span multiple hardware resources.
In progress	The rule is currently being programmed into the hardware.
Inactive	The rule is inactive and is not programmed in the hardware. This is typically seen when the hardware resources limit is reached.

## Examples

The following example displays all interfaces on which a specific MAC ACL is applied in the incoming direction. The ACL named macFoo is applied on interface 1/2/1 to filter incoming routed traffic only.

```
device# show access-list mac macFoo in
mac routed access-list macFoo on TenGigabitEthernet 1/2/1 at Ingress (From User)
  seq 10 permit host 0000.0101.0214 count (Active)
  seq 20 deny any count (Active)
```

The following example displays the names of IPv4 ACLs on device interfaces to which they are applied, the incoming/outgoing direction, and if **switched** or **routed** was specified.

```
device# show access-list ip
Interface Ve 171
  Inbound access-list is not set
  Outbound access-list is IPV4_ACL_000 (From User)
Interface TenGigabitEthernet 1/2/1
  Inbound switched access-list is IP_ACL_STD_EXAMPLE (From User)
  Outbound access-list is IP_ACL_EXT_EXAMPLE (From User)
```

The following example displays all interfaces on which an IPv4 ACL is applied in the outgoing direction.

```
device# show access-list ip IPV4_ACL_000 out
ip access-list IPV4_ACL_000 on Ve 171 at Egress (From User)
  seq 0 deny ip host 0.0.0.0 host 10.0.0.0 (Active)
```

The following example displays all interfaces on which an IPv6 ACL is applied in the incoming direction.

```
device# show access-list ipv6 distList in
ipv6 access-list distList on TenGigabitEthernet 122/1/2 at Ingress (From User)
  seq 10 deny 2001:125:132:35::/64 (Active)
  seq 20 deny 2001:54:131::/64 (Active)
  seq 30 deny 2001:5409:2004::/64 (Active)
  seq 40 permit any (Active)
```

The following example displays all ACLs applied on a specified interface in the incoming direction.

```
device# show access-list interface tengigabitethernet 1/4/1 in
ipv6 access-list ipv6-std-acl on TenGigabitEthernet 1/4/1 at Ingress (From User)
  seq 10 permit host 0:1::1 (Active)
  seq 20 deny 0:2::/64 (Active)
  seq 30 hard-drop any count (Active)
```

The following example displays details of ACLs applied in the outgoing direction to VE 121, RBridge 2.

```
device# show access-list interface ve 171 out rbridge-id 2
ip access-list IPV4_ACL_000 on Ve 171 at Egress (From User)
  seq 0 deny ip host 0.0.0.0 host 10.0.0.0 (Active)
```

The following example displays information for ACLs applied to a VXLAN overlay-gateway named gw121.

```
device# show access-list overlay-gateway gw121 in
mac access-list stdmacaclin on overlay-gateway gw121 at Ingress (From User)
  seq 11 permit 1111.1112.1113 7777.7777.7777 count log (Active)
  seq 12 permit 1111.1112.1114 7777.7777.7777 count log (Active)

ip access-list stdipaclin on overlay-gateway gw121 at Ingress (From User)
  seq 11 deny 11.22.33.44 255.255.255.0 count log (Active)
  seq 12 deny 11.22.33.45 255.255.255.0 count log (Active)

ipv6 access-list stdip6aclin on overlay-gateway gw121 at Ingress (From User)
  seq 20 deny any count log (Active)
```

The following example displays information for one or both of the Layer 3 receive-path ACLs applied to a specified RBridge.

```
device# show access-list rbridge-id 1 in
ipv6 access-list ipv6-receive-acl-example on System at Ingress on rbridge-id 1 (From Receive ACL)
  seq 10 hard-drop tcp host 10::1 any count (Active)
  seq 20 hard-drop udp any host 20::1 count (Active)
  seq 30 permit tcp host 10::2 any eq telnet count (Active)
  seq 40 permit tcp host 10::2 any eq bgp count (Active)
```

# show access-list-log buffer

Displays the contents of the log buffer for all ACLs, or for a specified interface.

## Syntax

```
show access-list-log buffer [ interface { <N>gigabitethernet rbridge_id/slot/port | port-channel index } ]
```

## Parameters

### interface

Filters by interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge\_id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

### port-channel *index*

Specifies a port-channel interface.

## Modes

Privileged EXEC mode

## Examples

The following example displays the contents of the log buffer for all ACLs.

```
device# show access-list-log buffer
Frames Logged on interface 1/2/1 :
-----
Frame Received Time : Fri Dec 9 3:8:48 2011
Ethernet,          Src : (00:34:56:78:0a:ab), Dst: (00:12:ab:54:67:da)
  Ethtype          : 0x8100
  Vlan tag type    : 0x800
  VlanID           : 0x1
Internet proto, Src : 192.85.1.2, Dst: 192.0.0.1
  Interface       :
  Type of service : 0
  Length          : 110
  Identification  : 0
  Fragmentation   : 00 00
  TTL             : 255
  protocol        : 253
  Checksum        : 39 3a
  Payload type    :
packet(s) repeated : 30
-----
```

## Related Commands

[access-group](#), [access-list](#), [show access-list-log buffer config](#)

# show access-list-log buffer config

Displays the configuration of the ACL buffer.

## Syntax

```
show access-list-log buffer config
```

## Modes

Privileged EXEC mode

## Examples

The following example displays the configuration of the ACL buffer.

```
device# show access-list-log buffer config
ACL Logging Enabled.
ACL logging Buffer configuration: Buffer type is circular and Buffer size is 1600.
```

## Related Commands

[access-group](#), [access-list](#), [show access-list-log buffer](#)

# show ag

Displays the current Access Gateway configuration on a switch.

## Syntax

```
show ag [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID for the switch.

## Modes

Privileged EXEC mode

## Usage Guidelines

The command displays current, active configuration information for Access Gateway, such as the switch identification, number and type of ports, enabled policies, port grouping, and attached fabric details

Consider these guidelines when automatic login balancing (lb) mode is enabled or disabled for a port group.

- When lb mode is disabled in a port group, the **show running-config ag**, **show ag map**, and **show ag** commands display the configured VF\_Port to N\_Port mapping. This is because configured and active mapping are the same.
- When LB mode is enabled in a port group, the **show ag** and **show ag map** commands display the active mapping only because VF\_Port to N\_Port mapping is based on the current distributed load across all N\_Ports. The **show running-config ag** command displays the configured mapping only.



## Examples

Following is an example of the Access Gateway configuration on RBridge 5:

```
sw0# show ag
Rbridge-ID 5:
-----
Name : sw0
NodeName : 10:00:00:05:33:f4:78:04
Number of Ports : 32
IP Address(es) : 10.37.209.80
Firmware Version : v4.1.0pgoel_pit02_nos4_1_10_10
Number of N_Ports(Fi) : 2
Number of VF_Ports : 0
Policies Enabled : pg
Persistent ALPA : Disabled
Port Group information :
PG_ID PG_Name PG_Mode PG_Members
-----
0 pg0 lb 5/0/1, 5/0/2, 5/0/3, 5/0/4,
5/0/5, 5/0/6, 5/0/7, 5/0/8
-----
Fabric Information :
Attached Fabric Name N_Ports(Fi)
-----
10:00:00:05:33:72:f5:5a 5/0/1, 5/0/2
N_Port(Fi) information :
Port PortID Attached PWWN IP_Addr VF_Ports
-----
Fi 5/0/1 0x020200 20:02:00:05:33:72:f5:5a 10.37.209.86 None
798 Network OS Command Reference
53-1003226-01
5 show ag
Fi 5/0/2 0x020300 20:03:00:05:33:72:f5:5a 10.37.209.86 None
-----
VF_Port information :
VF_Port Eth_Port PortID Attached PWWN N_Port(Fi)
-----
None
-----
```

# show ag map

Displays the current VF\_Port mapping to N\_Ports in Access Gateway mode on a specific switch or on all switches in the VCS cluster.

## Syntax

```
show ag map nport [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*nport*

N\_Port number supported by the switch model in /rbridge-id/port group/N Port format.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

To display VF\_Ports currently mapped to N\_Ports on a specific switch, enter **show ag map rbridge-id** *rbridge-id*.

To display VF\_Ports currently mapped to a specific N\_Port on a specific switch, enter **show ag map nport rbridge-id** *rbridge-id*. In Network OS commands, N\_Ports are designated by the format rbridge-id/port group/N\_Port. Therefore, N\_Port 5/0/1 designates that N\_Port 1 resides in port group 0 on RBridge 5. The **show ag map** command for this N\_Port would be the following:

```
show ag map 5/0/1 rbridge-id 5
```

To display VF\_Ports currently mapped to a specific N\_Port on all switches in the VCS cluster, enter **show ag map nport rbridge-id** **all**.

Consider these guidelines when Automatic Login Balancing (LB) mode is enabled or disabled for a port group.

- When Login Balancing (LB) mode is disabled in a port group, the **show running-config ag**, **show ag map**, and **show ag** commands display the configured VF\_Port to N\_Port mapping. This is because configured and active mapping are the same.
- When LB mode is enabled in a port group, the **show ag** and **show ag map** commands display the active mapping only because VF\_Port to N\_Port mapping is based on the current distributed load across all N\_Ports. The **show running-config ag** command displays the configured mapping only.

## Examples

Displaying port mapping information for a switch.

```
sw0# show ag map rbridge 5
Rbridge-ID 5:
-----
N_Port(Fi) PG_ID PG_Name Current_VF_Ports
-----
5/0/1 0 pg0 None
5/0/2 0 pg0 None
5/0/3 0 pg0 None
5/0/4 0 pg0 None
5/0/5 0 pg0 None
800 Network OS Command Reference
53-1003226-01
5 show ag map
5/0/6 0 pg0 None
5/0/7 0 pg0 None
5/0/8 0 pg0 None
-----
```

# show ag nport-utilization

Displays Access Gateway N\_Port utilization information. You can display this information either for a specific RBridge or for all.

## Syntax

```
show ag nport-utilization [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

You can display N\_Port utilization information either for a specific RBridge or for all.

### *rbridge-id*

Specify an RBridge ID.

### all

Display N\_Port utilization information for N\_ports on all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Access Gateway mode must be enabled.

There is a **clear** form of this command, which clears utilization information of RBridges specified. Enabling a port also clears the information.

If an N\_port is a trunk slave port, no utilization information is displayed. Instead, the bandwidth is included in the master port of the trunk.

## Command Output

The **show ag nport-utilization** command displays the highest bandwidth utilization and associated timestamp associated with each N\_port.

## Examples

The following command displays utilization information for Access Gateway N\_ports on RBridge 1:

```
sw0# show ag nport-utilization rbridge-ID 1
-----
Name                : sw0
NodeName            : 10:00:00:05:1e:e5:6d:27
Number of Ports     : 128
IP Address(es)      : 10.17.31.92
Firmware Version    : v5.0.0d
Number of N_Ports(Fi) : 2
Number of VF_Ports  : 2
Policies Enabled    : pg
Disabled
N_Port(Fi) information :
  Port      PortID      Attached PWWN      IP_Addr      VF_Ports
-----
Fi 1/0/7    0xa90900  2f:00:00:05:1e:80:31:4f  10.17.31.169  1/1/1, 1/1/2
             highest bandwidth utilization of 11 % recorded at Wed Apr 30 14:07:42 2014
Fi 1/0/8    0xa90900  2f:00:00:05:1e:80:31:4f  10.17.31.169  None
             trunk slave. bandwidth/traffic added to trunk master
-----
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ag pg

Displays information on Port Grouping (PG) configured on a switch for Access Gateway (AG) mode.

## Syntax

```
show ag pg [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

The command output includes N\_Ports and VF\_Ports in the group and Port Grouping (PG) modes enabled.

Access Gateway mode must be enabled.

## Examples

Following is an example showing port grouping information for RBridge 5:

```
sw0# show ag pg rbridge-id 5
Rbridge-ID 5:
```

```
-----
PG_ID  PG_Name  PG_Mode      N_Ports (Fi)      VF_Ports
-----
    0    pg0      lb           5/0/1,5/0/2,5/0/3,5/0/4,  1/5/1, 1/5/2, 1/5/3,
1/5/4,
                                           5/0/5,5/0/6,5/0/7,5/0/8  1/5/5, 1/5/6, 1/5/7,
1/5/8,
                                           1/5/9, 1/5/10, 1/5/11, 1/5/12,
                                           1/5/13, 1/5/14, 1/5/15, 1/5/16,
                                           1/5/17, 1/5/18, 1/5/19, 1/5/20,
                                           1/5/21, 1/5/22, 1/5/23, 1/5/24,
                                           1/5/25, 1/5/26, 1/5/27, 1/5/28,
                                           1/5/29, 1/5/30, 1/5/31, 1/5/32,
                                           1/5/33, 1/5/34, 1/5/35, 1/5/36,
                                           1/5/37, 1/5/38, 1/5/39, 1/5/40,
                                           1/5/41, 1/5/42, 1/5/43, 1/5/44,
                                           1/5/45, 1/5/46, 1/5/47, 1/5/48,
                                           1/5/49, 1/5/50, 1/5/51, 1/5/52,
                                           1/5/53, 1/5/54, 1/5/55, 1/5/56,
                                           1/5/57, 1/5/58, 1/5/59, 1/5/60,
1/5/61, 1/5/62, 1/5/63, 1/5/64
-----
```

# show arp

Displays the Address Resolution Protocol (ARP) entries.

## Syntax

```
show arp [ rbridge-id { all | rbridge-id } ]
show arp vrf name [ rbridge-id { all | rbridge-id } ]
show arp <N>gigabitethernet rbridge-id / slot / port [ vrf name ]
show arp port-channel number [ vrf name ]
show arp ve vlan-id [ vrf name ] [ rbridge-id { all | rbridge-id } ]
show arp ip ip-address [ vrf name ] [ rbridge-id { all | rbridge-id } ]
show arp slot slot-no [ ip ip-address ] [ vrf name ]
show arp [ dynamic | static ] [ summary ] [ vrf name ] [ rbridge-id { all | rbridge-id } ]
show arp access-list [ access-list-name ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

### vrf name

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel number

Specifies a port-channel interface. The range is from 1 through 6144.

### ve vlan-id

Specifies a virtual Ethernet (VE) interface.



**ip** *ip-address*

Specifies a next-hop IP address.

**slot** *slot-no*

Displays ARP information for a specified slot.

**dynamic**

Displays all the dynamic ARP entries in the ARP table.

**static**

Displays all the static ARP entries in the ARP table.

**summary**

Displays a summary of the ARP table.

**access-list** *access-list-name*

Displays information regarding dynamic ARP inspection (DAI) ACLs. You can optionally specify an ACL.

## Modes

Privileged EXEC mode

## Command Output

The **show arp** command displays the following information:

Output field	Description
Address	Displays the IP address.
Mac-address	Displays the MAC address or "UnResolved".
Interface	Displays the physical or VE interface.
MacResolved	Indicates if the record is listed in the MAC-address table. For a physical interface, displays "yes." For a VE, displays "yes" or "no".
Age	In hh:mm:ss format, displays the time since the most recent renewal of a dynamic entry. For a static entry, displays "Never".
Type	Displays "Dynamic", "Static", or "PreArp". ("PreArp" is ARP triggered other than by the data traffic, for example, by the static route.)

## Usage Guidelines

If leaked subnet routes are present, that information displays in the output.

## Examples

The following example displays the output of the basic **show arp** command.

```
device# show arp
Total ARPs : 1
Address      Mac-address      Interface  MacResolved  Type      Hardware
10.1.1.3     0010.9400.0001  Ve 100    yes          Dynamic    Yes

(Output truncated for brevity)
```

The following example displays the output of the **show arp slot** option.

```
device# show arp slot 0
Total ARPs : 15635
Address      Mac-address      Interface      MacResolved    Type           Hardware
10.10.11.2   0005.3379.5caa   Ve 811         yes            Dynamic       Yes
10.15.1.2    0019.0100.0001   Ve 15          yes            Dynamic       Yes
10.15.1.3    0019.0100.0002   Ve 15          yes            Dynamic       Yes
10.15.1.4    0019.0100.0003   Ve 15          yes            Dynamic       Yes
10.15.1.5    0019.0100.0004   Ve 15          yes            Dynamic       Yes
10.15.1.6    0019.0100.0005   Ve 15          yes            Dynamic       Yes
10.15.1.7    0019.0100.0006   Ve 15          yes            Dynamic       Yes
10.15.1.8    0019.0100.0007   Ve 15          yes            Dynamic       Yes
10.15.1.9    0019.0100.0008   Ve 15          yes            Dynamic       Yes
10.15.1.10   0019.0100.0009   Ve 15          yes            Dynamic       Yes
```

(Output truncated for brevity)

The following example displays the output of the **show arp summary** option.

```
device# show arp summary
No of Total Entries   No of Static Entries   No of Dynamic Entries   No of Leaked Entries   No of Pre-
arp Entries
-----
-----
3                     0                       1                       1                       1
```

The following example displays the output of the **show arp access-list** option.

```
device# show arp access-list
ARP access list arp-accesslist-filter
  permit ip host 10.10.1.85 mac host 0000.0000.0002
  permit ip host 10.10.1.88 mac host 0005.3326.a388 log
  permit ip host 10.10.1.89 mac host 0005.3326.44b3
ARP access list arpacl1
  permit ip host 10.10.1.85 mac host 0000.0000.0002
  permit ip host 10.10.1.88 mac host 0005.3326.a388 log
  permit ip host 10.10.1.89 mac host 0005.3326.44b3
  permit ip host 10.10.1.89 mac host 0005.3326.44be
  permit ip host 10.10.10.1 mac host 0002.0002.0002
  permit ip host 10.10.51.20 mac host 0010.9400.0005
  permit ip host 10.20.237.5 mac host 0000.2586.3652 log
  permit ip host 20.0.0.2 mac host 0002.0002.0004 log
ARP access list arpacl2
  permit ip host 10.10.41.10 mac host 0000.0100.0002 log
  permit ip host 10.10.41.10 mac host 0010.9400.0002 log
  permit ip host 10.10.51.10 mac host 0010.9400.0002 log
  permit ip host 10.10.51.10 mac host 0010.9400.0003
```

## History

Release version	Command history
7.0.1	This command was modified to display leaked subnet routes.
7.1.0	The <b>show arp summary</b> form of this command was modified to display pre-arp entries and leaked arp entries.

# show arp access-list

Displays one or all ARP access lists (ACLs) available on a device, including **permit** statements.

## Syntax

```
show arp access-list [ acl-name ]
```

## Parameters

*acl-name*

Specifies the name of an ARP ACL defined on the device.

## Modes

Privileged EXEC mode

## Examples

The following example displays the name and permit statements of an ARP ACL named "list1".

```
device# show arp access-list list1
ARP access list list1
  permit ip host 192.85.1.2 mac host 0010.9400.0002
  permit ip host 192.85.1.3 mac host 0010.9400.0003 log
  permit ip host 196.2.1.2 mac host 0020.3200.0008
```

The following example displays the name and permit statements of all ARP ACLs.

```
device# show arp access-list
ARP access list list1
  permit ip host 192.85.1.2 mac host 0010.9400.0002
  permit ip host 192.85.1.3 mac host 0010.9400.0003 log
  permit ip host 196.2.1.2 mac host 0020.3200.0008
ARP access list list2
  permit ip host 20.20.20.1 mac host 0011.9400.0001 log
  permit ip host 30.30.30.1 mac host 0011.9400.0002
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show bare-metal

Displays the current bare-metal state. Bare-metal configuration can enable a switch to join a VCS cluster by means of plug and play.

## Syntax

```
show bare-metal
```

## Modes

Privileged EXEC mode

## Command Output

The **show bare-metal** command displays the following information:

Output field	Description
disable	The switch is not available for bare-metal configuration.
enable	The switch is available for bare-metal configuration.

## Examples

The following example illustrates a switch on which bare-metal configuration is disabled.

```
device# show bare-metal
state : disable
```

The following example illustrates a switch on which bare-metal configuration is enabled.

```
device# show bare-metal
state : enable
```

## History

Release version	Command history
6.0.1	This command was introduced.

## Related Commands

[firmware download](#), [preprovision](#), [write erase](#)

# show bfd

Displays Bidirectional Forwarding Detection (BFD) information.

## Syntax

```
show bfd [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies a RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show bfd** command displays the following information on a stackable switch:

Output field	Description
BFD State	Specifies whether BFD is enabled or disabled on the device.
Version	Specifies the version of the BFD protocol operating on the device.
Supported Protocols	Specifies the protocols that are registered for the particular session.
All Sessions	
Current	The number of BFD sessions currently operating on the device.
Max Allowed	The maximum number of BFD sessions that are allowed on the device. The maximum number of sessions supported on a device is 250.
Max Exceeded Count	The number of times the request to set up a BFD session was declined because it would have resulted in exceeding the maximum number of BFD sessions allowed on the device.
Port	The port that BFD is enabled on. Entry for a port will be displayed, only if it has at least one session on that interface.
MinTx	The interval in milliseconds between which the device desires to send a BFD message from this port to its peer.
MinRx	The interval in milliseconds that this device desires to receive a BFD message from its peer on this port.
Mult	The number of times that the device will wait for the MinRx time on this port before it determines that its peer device is non-operational.
Sessions	The number of BFD sessions originating on this port.

The **show bfd** command displays the following information on a chassis:

Output field	Description
BFD State	Specifies if BFD is enabled or disabled on the device.
Version	Specifies the version of the BFD protocol operating on the device.
Supported Protocols	Specifies the protocols that are registered for the particular session.
All Sessions	
Current	The number of BFD sessions currently operating on the device.
Max Allowed	The maximum number of BFD sessions that are allowed on the device. The maximum number of sessions supported on a device is 250.
Max Exceeded Count	The number of times the request to set up a BFD session was declined because it would have resulted in exceeding the maximum number of BFD sessions allowed on the device.
Agent sessions:	
Maximum Allowed on LC	The maximum number of BFD sessions that are allowed on an interface module. The maximum number of sessions supported on a Line Card Module is 40.
Maximum Exceeded count for LCs	The number of times the request to set up a BFD session was declined because it would have resulted in exceeding the maximum number of BFD sessions allowed on an interface module.
LC	The number of the interface module that the Current Session Count is displayed for.
Tx/Rx Sessions	The number of Transmit (Tx) and Receive (Rx) BFD sessions currently operating on the specified interface module.
BFD Enabled ports count	The number of ports on the device that has been enabled for BFD.
Port	The port that BFD is enabled on. Entry for a port will be displayed, only if it has at least one session on that interface.
MinTx	The interval in milliseconds between which the device desires to send a BFD message from this port to its peer.
MinRx	The interval in milliseconds that this device desires to receive a BFD message from its peer on this port.
Mult	The number of times that the device will wait for the MinRx time on this port before it determines that its peer device is non-operational.
Sessions	The number of BFD sessions originating on this port.

## Examples

The following example shows sample output from the **show bfd** command on a stackable switch.

```
device# show bfd

Rbridge-id:1
  BFD State: ENABLED, Version: 1
  Supported Protocols: static-ip, tunnel, ospf6, ospf
  All Sessions: Current: 6 Max Allowed: 250 Max Exceeded Count: 0

  Port      MinTx      MinRx      Mult Sessions
  ====      =====      =====      =====
  Te 1/0/9   50         50         3      1
  Te 1/0/10  50         50         3      1
  Tu 61441   1000      1000      3      1
```

The following example shows sample output from the **show bfd** command on a chassis when the **rbridge-id all** keywords are used.

```
device# show bfd rbridge-id all
Rbridge-id:1
  BFD State: ENABLED, Version: 1
  Supported Protocols: static-ip, tunnel
  All Sessions: Current: 65 Max Allowed: 250 Max Exceeded Count: 0

  Port      MinTx      MinRx      Mult Sessions
  =====
  Te 1/0/4  50         50         3    65

Rbridge-id:2
  BFD State: ENABLED, Version: 1
  Supported Protocols: static-ip, tunnel
  All Sessions: Current: 65 Max Allowed: 250 Max Exceeded Count: 0

  Port      MinTx      MinRx      Mult Sessions
  =====
  Te 2/0/4  50         50         3    65
```

The following example shows sample output from the **show bfd** command on a stackable switch when the **rbridge-id** keyword is used and a RBridge is specified.

```
device# show bfd rbridge-id 1
Rbridge-id:1
  BFD State: ENABLED, Version: 1
  Supported Protocols: static-ip, tunnel, ospf6, ospf
  All Sessions: Current: 6 Max Allowed: 250 Max Exceeded Count: 0

  Port      MinTx      MinRx      Mult Sessions
  =====
  Te 1/0/9   50         50         3    1
  Te 1/0/10 50         50         3    1
  Tu 61441  1000       1000       3    1
```

The following example shows sample output from the **show bfd** command on a chassis.

```
device# show bfd
RBridge: 1
  BFD State: ENABLED, Version: 1
  Supported Protocols: ospf, Tunnel
  All Sessions: Current: 4 Max Allowed: 100 Max Exceeded Count: 0
  Agent Sessions: Max Allowed on LC: 40 Max Exceeded Count for LCs: 0

  LC  Tx/Rx Sessions  LC  Tx/Rx Sessions  LC  Tx/Rx Sessions  LC  Tx/Rx Sessions
  1   4/4              2   2/2              3   0/0              4   0/0
  5   0/0              6   0/0              7   0/0              8   0/0

  BFD Enabled ports count: 2
  Port      MinTx      MinRx      Mult Sessions
  Te 1/2/1  100       100       3    2
  Tunnel 1  100       100       3    2
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show bfd neighbors

Displays Bidirectional Forwarding Detection (BFD) neighbor information.

## Syntax

```
show bfd neighbors [ rbridge-id { rbridge-id | all }]
```

## Parameters

### rbridge-id

Specifies a RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Command output displays only sessions that are in default-vrf. To display non-default VRF entries, use the **show bfd neighbors vrf** command.

## Command Output

The **show bfd neighbors** command displays the following information:

Output field	Description
OurAddr	Specifies the source IP address of the interface on which this BFD session is running.
NeighAddr	The IPv4 or IPv6 address of the remote neighbor.
State	The current state of the BFD session: <ul style="list-style-type: none"> <li>• UP</li> <li>• DOWN</li> <li>• A.DOWN - The administrative down state.</li> <li>• INIT - The initialization state.</li> <li>• UNKNOWN - The current state is unknown.</li> </ul>
Int	Specifies the interface on which the BFD session is running.
Rbridge-id	Specifies the RBridge ID, where the particular session is running. In the case of tunnels, it will be the list of RBridges mentioned where the tunnel is configured.



## Examples

The following example shows sample output from the **show bfd neighbors** command.

```
device# show bfd neighbors

OurAddr      NeighAddr    State    Int          Rbridge-id
=====
10.0.6.1     10.0.6.2     Up       Te2/0/1
20.2.7.2     20.2.7.1     Up       Ve10         RB1
11.11.17.2   11.11.17.3   Down    Tunnel10     RB1-3,RB5
```

The following example shows sample output from the **show bfd neighbors** command when the **rbridge-id** keyword is used.

```
device# show bfd neighbors rbridge-id all

OurAddr      NeighAddr    State    Int          Rbridge-id
=====
11.1.1.1     11.1.1.21    DOWN    Te 1/0/4     1
11.1.1.1     11.1.1.22    DOWN    Te 1/0/4     1

21.1.1.1     21.1.1.21    DOWN    Te 2/0/4     2
21.1.1.1     21.1.1.22    DOWN    Te 2/0/4     2

31.1.1.1     31.1.1.21    DOWN    Te 3/0/4     3
31.1.1.1     31.1.1.22    DOWN    Te 3/0/4     3
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show bfd neighbors application

Displays Bidirectional Forwarding Detection (BFD) neighbor session information.

## Syntax

```
show bfd neighbors application { bgp | ospf | ospf6 | static-ip | tunnel [ details [ rbridge-id { rbridge-id | all } ] ] rbridge-id
  { rbridge-id | all } }
```

## Parameters

### bgp

Specifies Border Gateway Protocol (BGP) sessions.

### ospf

Specifies Open Shortest Path First (OSPF) sessions.

### ospf6

Specifies Open Shortest Path First version 3 (OSPFv3) sessions.

### static-ip

Specifies IP static route sessions.

### tunnel

Specifies a tunnel interface.

### details

Displays detailed neighbor information.

### rbridge-id

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show bfd neighbors application** command displays the following information:

Output field	Description
OurAddr	Specifies the source IP address of the interface on which this BFD session is running.
NeighAddr	The IPv4 or IPv6 address of the remote neighbor.
State	The current state of the BFD session: Up

Output field	Description
	Down A.DOWN - The administrative down state. INIT - The Init state. UNKNOWN - The current state is unknown.
Int	Specifies the interface on which the BFD session is running.
Rbridge-id	Specifies the RBridge ID, where the particular session is running. In the case of tunnels, it will be the list of RBridges mentioned where the tunnel is configured.

## Examples

The following example shows sample output from the **show bfd neighbors application** command when the **ospf** keyword is used.

```
device# show bfd neighbors application ospf

OurAddr      NeighAddr      State   Int           Rbridge-id
=====      =====      =====
5.0.0.1      5.0.0.2        UP      Te 1/0/10     1
2.0.0.1      2.0.0.2        UP      Te 1/0/9      1
```

The following example shows sample output from the **show bfd neighbors application** command when the **ospf** and **details** keywords are used.

```
device# show bfd neighbors application ospf details

OurAddr      NeighAddr      State   Int           Rbridge-id
=====      =====      =====
10.0.6.1     10.0.6.2       UP      Te2/0/1      RB1

Local        State: UP   Diag: 0   Demand mode: 0   Poll: 0
Received State: UP   Diag: 0   Demand mode: 0   Poll: 0   Final: 0
Local   MinTxInt(ms): 1000   MinRxInt(ms): 1000   Multiplier: 5
Received MinTxInt(ms): 1000   MinRxInt(ms): 1000   Multiplier: 5
  Rx Count: 3806   Tx Count: 4308
  LD/RD: 10001/10001   Heard from Remote: Y
Current Registered Protocols: ospf
Uptime: 0 day 0 hour 0 min 0 sec 0 msec
```

The following example shows sample output from the **show bfd neighbors application** command when the **tunnel** and **details** keywords are used.

```
device# show bfd neighbors application tunnel details

OurAddr      NeighAddr      State   Int           Rbridge-id
=====      =====      =====
1.1.1.1      4.4.4.4        UP      Tu 61441      1

Local   State: UP           Diag: 0           Demand mode: 0   Poll: 0
Received State: UP           Diag: 0           Demand mode: 0   Poll: 0
Final: 0
Local   MinTxInt(ms): 1000   MinRxInt(ms): 1000   Multiplier: 3
Received MinTxInt(ms): 1000   MinRxInt(ms): 1000   Multiplier: 3
Rx Count: 1078           Tx Count: 1039
LD/RD: 10001/10001   Heard from Remote: Y
Current Registered Protocols: tunnel
Uptime: 0 day 0 hour 13 min 31 sec 32 msec
```

show bfd neighbors application

## History

Release version	Command history
6.0.1	This command was introduced.

# show bfd neighbors dest-ip

Displays Bidirectional Forwarding Detection (BFD) neighbor information about destination devices.

## Syntax

```
show bfd neighbors dest-ip { ip address | ipv6 address } details [ rbridge-id { rbridge-id | all } ]
```

```
show bfd neighbors dest-ip { ip address | ipv6 address } interface { <N>gigabitethernet rbridge-id/slot/port | ve vlan_id }
```

```
show bfd neighbors dest-ip [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*ip address*

Specifies the IP address of the destination device.

*ipv6 address*

Specifies the IPv6 address of the destination device.

**details**

Displays detailed neighbor information about the destination device.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**interface**

Displays BFD neighbor interface information.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**ve vlan\_id**

Specifies a virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Command Output

The **show bfd neighbors dest-ip** command displays the following information:

Output field	Description
OurAddr	Specifies the source IP address of the interface on which this BFD session is running.
NeighAddr	The IPv4 or IPv6 address of the remote neighbor.
State	The current state of the BFD session: Up Down A.DOWN - The administrative down state. INIT - The Init state. UNKNOWN - The current state is unknown.
Int	Specifies the interface on which the BFD session is running.
Rbridge-id	Specifies the RBridge ID, where the particular session is running. In the case of tunnels, it will be the list of RBridges mentioned where the tunnel is configured.

## Examples

The following example shows sample output from the **show bfd neighbors dest-ip** command.

```
device# show bfd neighbors dest-ip 1.1.1.6
OurAddr      NeighAddr    State   Int          Rbridge-id
=====
1.1.1.5      1.1.1.6     UP      Ve 5         5
```

The following example shows sample output from the **show bfd neighbors dest-ip** command when the **details** keyword is used.

```
device# show bfd neighbors dest-ip 1.1.1.6 details
OurAddr      NeighAddr    State   Int          Rbridge-id
=====
1.1.1.5      1.1.1.6     UP      Ve 5         5

Local      State: UP          Diag: 0          Demand mode: 0   Poll: 0
Received State: UP          Diag: 0          Demand mode: 0   Poll: 0   Final: 1
Local      MinTxInt(ms): 50  MinRxInt(ms): 50 Multiplier: 3
Received MinTxInt(ms): 50  MinRxInt(ms): 50 Multiplier: 3
Rx Count: 226354          Tx Count: 234323
LD/RD:      1/11          Heard from Remote: Y
Current Registered Protocols: ospf
Uptime: 0 day 3 hour 4 min 48 sec 248 msec
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show bfd neighbors details

Displays detailed Bidirectional Forwarding Detection (BFD) neighbor information.

## Syntax

```
show bfd neighbors details [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies a RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show bfd neighbors details** command displays the following information:

Output field	Description
OurAddr	Specifies the source IP address of the interface on which this BFD session is running.
NeighAddr	Specifies the IPv4 address of the remote neighbor.
State	Specifies the current state of the BFD session: <ul style="list-style-type: none"> <li>• UP</li> <li>• DOWN</li> <li>• A.DOWN - The administrative down state.</li> <li>• INIT - The initialization state.</li> <li>• UNKNOWN - The current state is unknown.</li> </ul>
Int	Specifies the interface on which the BFD session is running.
Rbridge-id	Specifies the RBridge ID, where the particular session is running. In the case of tunnels, it will be the list of RBridges mentioned where the tunnel is configured.
Local:	
State	State of the local device.
Diag	Value of the diagnostic field in the BFD control message as used by the device in the last message sent.
Demand mode	Value of the demand in the BFD control message as used by the device in the last message received.
Poll	Value of the poll in the BFD control message as used by the device in the last message sent or received.
Received	

Output field	Description
State	State of the remote device.
Diag	Value of the diagnostic field in the BFD control message as used by the device in the last message received.
Demand mode	Value of the demand in the BFD control message as used by the device in the last message received.
Poll	Value of the poll in the BFD control message as used by the device in the last message received.
Final	Value of the final bit in the BFD control message as used by the device in the last message received.
Local	The local device
MinTxInt(ms)	The interval in milliseconds between which the device will send a BFD message from this local neighbor port to its peer.
MinRxIntMinTxInt(ms)	The interval in milliseconds that the neighbor device waits to receive a BFD message from its peer on this local port.
Multiplier	The number of times that the neighbor device will wait for the MinRxInterval time on this port before it determines that its peer device is non-operational.
Received	
MinTxInt(ms)	The interval in milliseconds between which the device will send a BFD message from the remote neighbor port to its peer.
MinRxIntMinTxInt(ms)	The interval in milliseconds that the neighbor device waits to receive a BFD message from its peer on this remote port.
Multiplier	The number of times that the remote neighbor device will wait for the MinRxInterval time on this port before it determines that its peer device is nonoperational.
Rx Count	Total number of BFD control messages received from the remote peer.
Tx Count	Total number of BFD control messages sent to the remote peer.
LD/RD	Local and remote descriptor
Heard from Remote	Indicates remote BFD neighbor has been heard.
Current Registered Protocols	Specifies the protocols that are registered for the particular session.
Uptime	The amount of time the BFD session has been in the up state.



## Examples

The following example shows sample output from the **show bfd neighbors details** command when the **rbridge-id** keyword is used.

```
device# show bfd neighbors details rbridge-id all

OurAddr          NeighAddr        State   Int           Rbridge-id
=====          =====          =====
11.1.1.1         11.1.1.21        DOWN    Te 1/0/4      1

Local   State: DOWN           Diag: 0           Demand mode: 0    Poll: 0
Received State: A.DOWN   Diag: 0           Demand mode: 0    Poll: 0    Final: 0
Local   MinTxInt(ms): 50   MinRxInt(ms): 50 Multiplier: 3
Received MinTxInt(ms): 0 MinRxInt(ms): 0   Multiplier: 0
Rx Count: 0              Tx Count: 0
LD/RD:      1/0         Heard from Remote: N
Current Registered Protocols: static-ip
Uptime: 0 day 0 hour 0 min 0 sec 0 msec

OurAddr          NeighAddr        State   Int           Rbridge-id
=====          =====          =====
11.1.1.1         11.1.1.22        DOWN    Te 1/0/4      1

Local   State: DOWN           Diag: 0           Demand mode: 0    Poll: 0
Received State: A.DOWN   Diag: 0           Demand mode: 0    Poll: 0    Final: 0
Local   MinTxInt(ms): 50   MinRxInt(ms): 50 Multiplier: 3
Received MinTxInt(ms): 0 MinRxInt(ms): 0   Multiplier: 0
Rx Count: 0              Tx Count: 0
LD/RD:      2/0         Heard from Remote: N
Current Registered Protocols: static-ip
Uptime: 0 day 0 hour 0 min 0 sec 0 msec

OurAddr          NeighAddr        State   Int           Rbridge-id
=====          =====          =====
21.1.1.1         21.1.1.21        DOWN    Te 2/0/4      2

Local   State: DOWN           Diag: 0           Demand mode: 0    Poll: 0
Received State: A.DOWN   Diag: 0           Demand mode: 0    Poll: 0    Final: 0
Local   MinTxInt(ms): 50   MinRxInt(ms): 50 Multiplier: 3
Received MinTxInt(ms): 0 MinRxInt(ms): 0   Multiplier: 0
Rx Count: 0              Tx Count: 0
LD/RD:      1/0         Heard from Remote: N
Current Registered Protocols: static-ip
Uptime: 0 day 0 hour 0 min 0 sec 0 msec

OurAddr          NeighAddr        State   Int           Rbridge-id
=====          =====          =====
21.1.1.1         21.1.1.22        DOWN    Te 2/0/4      2

Local   State: DOWN           Diag: 0           Demand mode: 0    Poll: 0
Received State: A.DOWN   Diag: 0           Demand mode: 0    Poll: 0    Final: 0
Local   MinTxInt(ms): 50   MinRxInt(ms): 50 Multiplier: 3
Received MinTxInt(ms): 0 MinRxInt(ms): 0   Multiplier: 0
Rx Count: 0              Tx Count: 0
LD/RD:      2/0         Heard from Remote: N
Current Registered Protocols: static-ip
Uptime: 0 day 0 hour 0 min 0 sec 0 msec
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show bfd neighbors interface

Displays Bidirectional Forwarding Detection (BFD) neighbor information about specified interfaces.

```
show bfd neighbors interface { <N>gigabitethernet rbridge-id/slot/port | tunnel number [ details ] }
```

```
show bfd neighbors interface { ve vlan_id [ details [ rbridge-id { rbridge-id | all } ] | rbridge-id { rbridge-id | all } ] }
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### rbridge-id

Specifies an RBridge ID.

#### slot

Specifies a valid slot number.

#### port

Specifies a valid port number.

### tunnel number

Specifies a tunnel interface.

### details

Displays detailed neighbor interface information.

### ve vlan\_id

Specifies a virtual Ethernet (VE) interface.

### rbridge-id

Specifies a RBridge or all RBridges.

#### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show bfd neighbors interface** command displays the following information:

Output field	Description
OurAddr	Specifies the source IP address of the interface on which this BFD session is running.
NeighAddr	The IPv4 or IPv6 address of the remote neighbor.
State	The current state of the BFD session:

Output field	Description
	Up Down A.DOWN - The administrative down state. INIT - The Init state. UNKNOWN - The current state is unknown.
Int	Specifies the interface on which the BFD session is running.
Rbridge-id	Specifies the RBridge ID, where the particular session is running. In the case of tunnels, it will be the list of RBridges mentioned where the tunnel is configured.

## Examples

This example shows sample output from the **show bfd neighbors interface** command when the **ve** keyword is used.

```
device# show bfd neighbors interface ve 5

OurAddr      NeighAddr      State   Int           Rbridge-id
=====
1.1.1.5      1.1.1.6        UP      Ve 5          5
```

This example shows sample output from the **show bfd neighbors interface** command when the **ve** and **details** keywords are used.

```
device# show bfd neighbors interface ve 5 details

OurAddr      NeighAddr      State   Int           Rbridge-id
=====
1.1.1.5      1.1.1.6        UP      Ve 5          5

Local State: UP          Diag: 0          Demand mode: 0   Poll: 0
Received State: UP      Diag: 0          Demand mode: 0   Poll: 0   Final: 1
Local MinTxInt(ms): 50  MinRxInt(ms): 50 Multiplier: 3
Received MinTxInt(ms): 50 MinRxInt(ms): 50 Multiplier: 3
Rx Count: 225447        Tx Count: 233383
LD/RD: 1/11            Heard from Remote: Y
Current Registered Protocols: ospf
Uptime: 0 day 3 hour 4 min 0 sec 208 msec
```

This example shows sample output from the **show bfd neighbors interface** command when the **tengigabitethernet** keyword is used.

```
device# show bfd neighbors interface tengigabitethernet 1/0/10

OurAddr      NeighAddr      State   Int           Rbridge-id
=====
5.0.0.1      5.0.0.2        UP      Te 1/0/10     1
```

This example shows sample output from the **show bfd neighbors interface** command when the **tengigabitethernet** and **details** keywords are used.

```
device# show bfd neighbors interface tengigabitethernet 1/0/10 details
OurAddr      NeighAddr      State      Int      Rbridge-id
=====      =====
5.0.0.1      5.0.0.2        UP         Te 1/0/10   1

Local      State: UP          Diag: 0          Demand mode: 0   Poll: 0
Received State: UP          Diag: 0          Demand mode: 0   Poll: 0
Final: 1
Local      MinTxInt(ms): 50   MinRxInt(ms): 50   Multiplier: 3
Received MinTxInt(ms): 50   MinRxInt(ms): 50   Multiplier: 3
Rx Count: 286316          Tx Count: 297032
LD/RD:      1/1          Heard from Remote: Y
Current Registered Protocols: ospf
Uptime: 0 day 4 hour 8 min 49 sec 40 msec
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show bfd neighbors vrf

Displays Bidirectional Forwarding Detection (BFD) neighbor information for specified VRF instances.

## Syntax

```
show bfd neighbors vrf vrfname [ details [ rbridge-id { rbridge-id | all } ] ] rbridge-id { rbridge-id | all }
```

## Parameters

*vrfname*

Specifies the name of the VRF instance.

**details**

Displays detailed neighbor information for a specified VRF.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show bfd neighbors vrf** command displays the following information:

Output field	Description
OurAddr	Specifies the source IP address of the interface on which this BFD session is running.
NeighAddr	The IPv4 or IPv6 address of the remote neighbor.
State	The current state of the BFD session: Up Down A.DOWN - The administrative down state. INIT - The Init state. UNKNOWN - The current state is unknown.
Int	Specifies the interface on which the BFD session is running.
Rbridge-id	Specifies the RBridge ID, where the particular session is running. In the case of tunnels, it will be the list of RBridges mentioned where the tunnel is configured.

## Examples

This example shows sample output from the **show bfd neighbors vrf** command.

```
device# show bfd neighbors vrf default-vrf

OurAddr          NeighAddr        State   Int           Rbridge-id
=====          =====          =====  ===           =====
1.1.1.5          1.1.1.6          UP      Ve 5          5
```

This example shows sample output from the **show bfd neighbors vrf** command when the **details** keyword is used.

```
device# show bfd neighbors vrf default-vrf details

OurAddr          NeighAddr        State   Int           Rbridge-id
=====          =====          =====  ===           =====
1.1.1.5          1.1.1.6          UP      Ve 5          5

Local   State: UP           Diag: 0           Demand mode: 0   Poll: 0
Received State: UP   Diag: 0           Demand mode: 0   Poll: 0   Final: 1
Local   MinTxInt(ms): 50    MinRxInt(ms): 50  Multiplier: 3
Received MinTxInt(ms): 50  MinRxInt(ms): 50  Multiplier: 3
Rx Count: 228465           Tx Count: 236512
LD/RD:      1/11           Heard from Remote: Y
Current Registered Protocols: ospf
Uptime: 0 day 3 hour 6 min 38 sec 704 msec
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show bgp evpn dampened-routes

Displays information about dampened MAC routes for BGP EVPN.

## Syntax

```
show bgp evpn dampened-routes
```

## Modes

Privileged EXEC mode

## Examples

The following example shows information about dampened MAC routes.

```
device# show bgp evpn dampened-routes
      Prefix                               Reuse-in
                                         (secs)
1     MAC: [10] [0011.1111.1174]         never
2     MAC: [10] [0011.1111.1175]         300
3     MAC: [10] [0011.1111.1176]         270
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.0.1	Command output was modified to reflect reuse time.

show bgp evpn interface port-channel

# show bgp evpn interface port-channel

Displays BGP EVPN information for a port-channel interface.

## Syntax

show bgp evpn interface port-channel *number*

## Parameters

*number*

Specifies a port-channel number. Valid values range from 1 through 6144.

## Modes

Privileged EXEC mode

## Examples

The following example shows BGP EVPN information for port-channel interfaces.

```
device# show bgp evpn interface port-channel
```

```
-----  
Name           IfIndex  ESI                               Status  ESI_Derive  Originated  
-----  
po26           2800001a 00.000000000000000000000000  UP      None        No  
po36           28000024 00.000000000000000000000000  UP      None        No
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show bgp evpn interface tunnel

Displays BGP EVPN information for a tunnel interface.

## Syntax

```
show bgp evpn interface tunnel [ ip address | dynamic-discovered ]
```

## Parameters

*ip address*

Specifies the IP address of the destination tunnel.

**dynamic-discovered**

Specifies dynamically discovered tunnels.

## Modes

Privileged EXEC mode

## Examples

The following example shows BGP EVPN information for tunnel interfaces.

```
device# device# show bgp evpn interface tunnel
```

```
VTEP ID      : 1
VRF ID       : 1
VTEP Source IP : 0x06000006
```

```
-----
IfIndex      Dest IP      Status      Type      MAC Learning on Tunnel
-----
0x7c00f001   12.0.0.12   Up          DYNAMIC YES
0x7c00f002   10.11.11.10 Up          DYNAMIC YES
0x7c00f003   78.0.0.78   Up          DYNAMIC YES
0x7c00f004   9.0.0.9     Up          DYNAMIC YES
0x7c00f005   40.0.0.40   Up          DYNAMIC YES
0x7c00f006   13.0.0.13   Up          DYNAMIC YES
0x7c00f007   50.0.0.50   Up          DYNAMIC YES
-----
```

## History

Release version	Command history
7.0.0	This command was introduced.

show bgp evpn l2route detail

# show bgp evpn l2route detail

Displays detailed information for BGP EVPN routes in the MAC-VRF table.

## Syntax

`show bgp evpn l2route detail`

## Modes

Privileged EXEC mode

## Examples

The following example shows detailed BGP EVPN information for Layer 2 routes.

```

device# show bgp evpn l2route detail

Total number of BGP EVPN Routes : 102247
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:1
1      Prefix: ARP:[10011][0000.abba.baba]:[IPv4:11.1.1.254], Status: BL,
Age
22h14m21s
      NEXT_HOP: 0.0.0.0, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
      AS_PATH:
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:
00:0
0:00:00:00:08
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 10011 L3_vni: 0
      ESI : 00.00000000000000000000
2      Prefix: ARP:[10011][0000.abba.baba]:[IPv4:11.1.1.254], Status: E,
Age:
22h13m43s
      NEXT_HOP: 10.11.11.10, Learned from Peer: 3.0.0.3 (2)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 2 1 3
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:
00:0
0:00:00:00:08 RT 3:10011
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 10011 L3_vni: 0
      ESI : 00.00000000000000000000
3      Prefix: ARP:[10011][0000.abba.baba]:[IPv4:11.1.1.254], Status: E, Age: 22h13m44s
      NEXT_HOP: 12.0.0.12, Learned from Peer: 3.0.0.3 (2)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 2 1 3
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT 3:10011
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 10011 L3_vni: 0
      ESI : 00.00000000000000000000
4      Prefix: ARP:[10011][0000.abba.baba]:[IPv4:11.1.1.254], Status: E, Age: 22h13m46s
      NEXT_HOP: 78.0.0.78, Learned from Peer: 2.0.0.2 (2)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 2 3
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT 3:10011
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 10011 L3_vni: 0
      ESI : 00.00000000000000000000
5      Prefix: ARP:[10011][0010.9400.0041]:[IPv4:11.1.1.8], Status: BE, Age: 18h38m52s
      NEXT_HOP: 10.11.11.10, Learned from Peer: 3.0.0.3 (2)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 2 1 3
        Extended Community: ExtCom:06:00:00:00:00:00:00:01 ExtCom:03:0c:00:00:00:00:00:08 RT 3:10011
      Mac Mobility Ext Community: 1
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 10011 L3_vni: 0
      ESI : 00.00000000000000000000
6      Prefix: ARP:[10011][0010.9400.00c1]:[IPv4:11.1.1.12], Status: BE, Age: 18h38m52s
      NEXT_HOP: 10.11.11.10, Learned from Peer: 3.0.0.3 (2)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 2 1 3
        Extended Community: ExtCom:06:00:00:00:00:00:00:01 ExtCom:03:0c:00:00:00:00:00:08 RT 3:10011
      Mac Mobility Ext Community: 1
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)

```

show bgp evpn l2route detail

```
L2_vni: 10011 L3_vni: 0  
ESI : 00.000000000000000000
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route next-hop

Displays information about the BGP EVPN routes with the specified next-hop in the MAC-VRF table.

## Syntax

```
show bgp evpn l2route next-hop { ip address | ipv6 address } type { arp | auto-discovery | ethernet-segment | inclusive-multicast | mac | nd }
```

## Parameters

*ip address*

Specifies the IP address.

*ipv6 address*

Specifies the IPv6 address.

**type**

Specifies the type of Layer 2 route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segments (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn l2route next-hop** command when the **arp** keyword is used and displays only the ARP routes with the specified next-hop.

```
device# show bgp evpn l2route next-hop 18.0.0.18 type arp

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix          Next Hop          MED          LocPrf        Weight Status
1  ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    18.0.0.18          0            100           0            E
    AS_PATH: 65002 65012
    L2_vni: 23 L3_vni: 0
    ESI : 00.00000000000000000000
2  ARP:[24][0000.abba.baba]:[IPv4:2.24.1.254]
    18.0.0.18          0            100           0            E
    AS_PATH: 65002 65012
    L2_vni: 24 L3_vni: 0
    ESI : 00.00000000000000000000
3  ARP:[25][0000.abba.baba]:[IPv4:2.25.1.254]
    18.0.0.18          0            100           0            E
    AS_PATH: 65002 65012
    L2_vni: 25 L3_vni: 0
    ESI : 00.00000000000000000000
4  ARP:[26][0000.abba.baba]:[IPv4:2.26.1.254]
    18.0.0.18          0            100           0            E
    AS_PATH: 65002 65012
    L2_vni: 26 L3_vni: 0
    ESI : 00.00000000000000000000
5  ARP:[27][0000.abba.baba]:[IPv4:2.27.1.254]
    18.0.0.18          0            100           0            E
    AS_PATH: 65002 65012
    L2_vni: 27 L3_vni: 0
    ESI : 00.00000000000000000000
6  ARP:[28][0000.abba.baba]:[IPv4:2.28.1.254]
    18.0.0.18          0            100           0            E
    AS_PATH: 65002 65012
    L2_vni: 28 L3_vni: 0
    ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn l2route next-hop** command when the **mac** keyword is used and displays only the MAC routes with the specified next-hop.

```
device# show bgp evpn l2route next-hop 18.0.0.18 type mac

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight  Status
1  MAC:[23][0000.abba.abba]
    18.0.0.18      0        100         0        E
    AS_PATH: 65002 65012
    L2_vni: 23
    ESI : 00.00000000000000000000
2  MAC:[23][0000.abba.baba]
    18.0.0.18      0        100         0        E
    AS_PATH: 65002 65012
    L2_vni: 23
    ESI : 00.00000000000000000000
3  MAC:[23][50eb.1a14.080b]
    18.0.0.18      0        100         0        BE
    AS_PATH: 65002 65012
    L2_vni: 23
    ESI : 00.00000000000000000000
4  MAC:[24][0000.abba.abba]
    18.0.0.18      0        100         0        E
    AS_PATH: 65002 65012
    L2_vni: 24
    ESI : 00.00000000000000000000
5  MAC:[24][0000.abba.baba]
    18.0.0.18      0        100         0        E
    AS_PATH: 65002 65012
    L2_vni: 24
    ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn l2route next-hop** command when the **inclusive-multicast** keyword is used and displays only the inclusive-multicast routes with the specified next-hop.

```
device# show bgp evpn l2route next-hop 18.0.0.18 type inclusive-multicast

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight  Status
1  IMR:[21][IPv4:18.0.0.18]
    18.0.0.18      0        100         0        BE
    AS_PATH: 65002 65012
    L2_vni: 21
2  IMR:[23][IPv4:18.0.0.18]
    18.0.0.18      0        100         0        BE
    AS_PATH: 65002 65012
    L2_vni: 23
3  IMR:[24][IPv4:18.0.0.18]
    18.0.0.18      0        100         0        BE
    AS_PATH: 65002 65012
    L2_vni: 24
4  IMR:[25][IPv4:18.0.0.18]
    18.0.0.18      0        100         0        BE
    AS_PATH: 65002 65012
    L2_vni: 25
5  IMR:[26][IPv4:18.0.0.18]
    18.0.0.18      0        100         0        BE
    AS_PATH: 65002 65012
    L2_vni: 26
...
```

show bgp evpn l2route next-hop

## History

Release version	Command history
7.0.0	This command was introduced.



# show bgp evpn l2route summary

Displays summary information for BGP EVPN routes in the MAC-VRF table.

## Syntax

```
show bgp evpn l2route summary
```

## Modes

Privileged EXEC mode

## Examples

The following example shows summary BGP EVPN information for Layer 2 routes.

```

device# show bgp evpn l2route summary

Total number of BGP EVPN routes (NLRIs) Installed      : 103381
Distinct BGP destination EVPN Routes                  : 102280
EVPN Routes originated by this router                  : 100265
EVPN Routes selected as BEST routes                    : 101816
EVPN Routes Installed as BEST routes                  : 101792
EVPN BEST routes not installed in IP forwarding table  : 488
Unreachable EVPN routes (no IGP route for NEXTHOP)   : 0
IBGP EVPN routes selected as best routes              : 0
EBGP EVPN routes selected as best routes              : 1527
BEST EVPN routes not valid for IP forwarding table    : 0

Distinct BGP destination ARP routes                    : 188

Filtered ARP routes                                    : 3958
ARP routes originated by this router                   : 33
ARP routes selected as BEST routes                    : 91
ARP routes Installed as BEST routes                   : 67
ARP BEST routes not installed in IP forwarding table   : 121
Unreachable ARP routes (no IGP route for NEXTHOP)    : 0
IBGP ARP routes selected as best routes               : 0
EBGP ARP routes selected as best routes               : 34
ARP BEST routes not valid for IP forwarding table     : 0

Distinct BGP destination ND routes                    : 291

Filtered ND routes                                     : 2932
ND routes originated by this router                    : 65
ND routes selected as BEST routes                     : 194
ND routes Installed as BEST routes                    : 194
ND BEST routes not installed in IP forwarding table   : 97
Unreachable ND routes (no IGP route for NEXTHOP)    : 0
IBGP ND routes selected as best routes                : 0
EBGP ND routes selected as best routes                : 129
ND BEST routes not valid for IP forwarding table     : 0

Distinct BGP destination MAC routes                   : 101541

Filtered MAC routes                                    : 4631
MAC routes originated by this router                   : 100131
MAC routes selected as BEST routes                    : 101345
MAC routes Installed as BEST routes                   : 101345
MAC BEST routes not installed in IP forwarding table  : 196
Unreachable MAC routes (no IGP route for NEXTHOP)   : 0
IBGP MAC routes selected as best routes               : 0
EBGP MAC routes selected as best routes               : 1214
MAC BEST routes not valid for IP forwarding table    : 0

Distinct BGP destination AD routes                   : 4

Filtered AD routes                                     : 0
AD routes originated by this router                    : 0
AD routes selected as BEST routes                     : 2
AD routes Installed as BEST routes                    : 2
AD BEST routes not installed in IP forwarding table   : 2
Unreachable AD routes (no IGP route for NEXTHOP)    : 0
IBGP AD routes selected as best routes                : 0
EBGP AD routes selected as best routes                : 2
AD BEST routes not valid for IP forwarding table     : 0

Distinct BGP destination IMR routes                   : 256

Filtered IMR routes                                    : 320
IMR routes originated by this router                   : 36
IMR routes selected as BEST routes                    : 184
IMR routes Installed as BEST routes                   : 184

```

```
IMR BEST routes not installed in IP forwarding table : 72
Unreachable IMR routes (no IGP route for NEXTHOP)   : 0
IBGP IMR routes selected as best routes             : 0
EBGP IMR routes selected as best routes             : 148
IMR BEST routes not valid for IP forwarding table    : 0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route type arp

Displays information about BGP EVPN address-resolution protocol (ARP) routes in the MAC-VRF table.

## Syntax

**show bgp evpn l2route type arp**

**show bgp evpn l2route type arp brief**

**show bgp evpn l2route type arp detail**

**show bgp evpn l2route type arp** *ip address mac mac address ethernet-tag tag-id*

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

*ip address*

Specifies the IP address.

**mac** *mac address*

Specifies a Media Access Control (MAC) address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn l2route type arp** command when no arguments or keywords are used.

```
device# show bgp evpn l2route type arp

Total number of BGP EVPN ARP Routes : 889
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop           MED           LocPrf        Weight Status
Route Distinguisher: 6.0.0.6:1
1      ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
      0.0.0.0           0             100           0             BL
      AS_PATH:
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
2      ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
      19.0.0.19          0             100           0             E
      AS_PATH: 65002 65110
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
3      ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
      76.0.0.76          0             100           0             E
      AS_PATH: 65002 65110
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
4      ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
      18.0.0.18          0             100           0             E
      AS_PATH: 65002 65012
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
5      ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
      76.0.0.76          0             100           0             E
      AS_PATH: 65002 65110
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
6      ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
      23.0.0.23          0             100           0             E
      AS_PATH: 65002 65078
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn l2route type arp** command when the **brief** keyword is used.

```

device# show bgp evpn l2route type arp brief

Total number of BGP EVPN ARP Routes : 889
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop          MED          LocPrf          Weight Path
Route Distinguisher: 6.0.0.6:1
*> ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    0.0.0.0          0          100          0          ?
* ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    19.0.0.19        100          0          65002 65110 ?
* ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    76.0.0.76        100          0          65002 65110 ?
* ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    18.0.0.18        100          0          65002 65012 ?
* ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    76.0.0.76        100          0          65002 65110 ?
* ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    23.0.0.23        100          0          65002 65078 ?
  ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254]
    23.0.0.23        100          0          65002 65078 ?
*> ARP:[24][0000.abba.baba]:[IPv4:2.24.1.254]
    0.0.0.0          0          100          0          ?
* ARP:[24][0000.abba.baba]:[IPv4:2.24.1.254]
    19.0.0.19        100          0          65002 65110 ?
* ARP:[24][0000.abba.baba]:[IPv4:2.24.1.254]
    76.0.0.76        100          0          65002 65110 ?
* ARP:[24][0000.abba.baba]:[IPv4:2.24.1.254]
    18.0.0.18        100          0          65002 65012 ?
* ARP:[24][0000.abba.baba]:[IPv4:2.24.1.254]
    76.0.0.76        100          0          65002 65110 ?
* ARP:[24][0000.abba.baba]:[IPv4:2.24.1.254]
    23.0.0.23        100          0          65002 65078 ?
...

```

The following example shows output for the **show bgp evpn l2route type arp** command when the **detail** keyword is used.

```
device# show bgp evpn l2route type arp detail

Total number of BGP EVPN ARP Routes : 889
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:1
1     Prefix: ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254], Status: BL, Age: 16h39m10s
      NEXT_HOP: 0.0.0.0, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
      AS_PATH:
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
        Default Extd Gw Community: Received
        Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
        L2_vni: 23 L3_vni: 0
        ESI : 00.00000000000000000000
2     Prefix: ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254], Status: E, Age: 16h37m40s
      NEXT_HOP: 19.0.0.19, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65110:23
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
3     Prefix: ARP:[23][0000.abba.baba]:[IPv4:2.23.1.254], Status: E, Age: 16h37m54s
      NEXT_HOP: 76.0.0.76, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65110:23
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route type auto-discovery

Displays information about BGP EVPN auto-discovery routes in the MAC-VRF table.

## Syntax

```
show bgp evpn l2route type auto-discovery
show bgp evpn l2route type auto-discovery brief
show bgp evpn l2route type auto-discovery detail
show bgp evpn l2route type auto-discovery esi-value ethernet-tag tag-id
```

## Parameters

**brief**  
Displays summary information.

**detail**  
Displays detailed information.

**esi-value**  
Specifies a 10 byte Ethernet Segment Identifier (ESI) value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH.HH).

**ethernet-tag *tag-id***  
Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn l2route type auto-discovery** command no arguments or keywords are used.

```
device# show bgp evpn l2route type auto-discovery

Total number of BGP EVPN AD Routes : 2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix          Next Hop          MED          LocPrf      Weight Status
Route Distinguisher: 9.0.0.9:1
1      AD:[01.006501e05200000a00] [4294967295]
       23.0.0.23          0              100          0           BME
       AS_PATH: 65002 65078
2      AD:[01.006501e05200000a00] [4294967295]
       23.0.0.23          0              100          0           ME
       AS_PATH: 65002 65078
```



The following example shows output for the **show bgp evpn l2route type auto-discovery** command when the **brief** keyword is used.

```
device# show bgp evpn l2route type auto-discovery brief

Total number of BGP EVPN AD Routes : 2
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network      Next Hop      MED      LocPrf      Weight Path
Route Distinguisher: 9.0.0.9:1
*> AD:[01.006501e05200000a00][4294967295]
      23.0.0.23              100      0      65002 65078 ?
*   AD:[01.006501e05200000a00][4294967295]
      23.0.0.23              100      0      65002 65078
```

The following example shows output for the **show bgp evpn l2route type auto-discovery** command when the **detail** keyword is used.

```
device# show bgp evpn l2route type auto-discovery detail

Total number of BGP EVPN AD Routes : 2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 9.0.0.9:1
1   Prefix: AD:[01.006501e05200000a00][4294967295], Status: BME, Age: 1d6h39m27s
      NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65078
      Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
      ESI Label Ext Community: 0      All-Active
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
2   Prefix: AD:[01.006501e05200000a00][4294967295], Status: ME, Age: 1d6h39m27s
      NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65078
      Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
      ESI Label Ext Community: 0      All-Active
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route type ethernet-segment

Displays information about BGP EVPN Ethernet Segment (ES) routes in the MAC-VRF table.

## Syntax

```
show bgp evpn l2route type ethernet-segment
```

```
show bgp evpn l2route type ethernet-segment brief
```

```
show bgp evpn l2route type ethernet-segment detail
```

```
show bgp evpn l2route type ethernet-segment esi-value value { ipv4-address address | ipv6-address address }
```

## Parameters

### brief

Displays summary information.

### detail

Displays detailed information.

### esi-value *value*

Specifies a 10 byte Ethernet Segment Identifier (ESI) value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH.HH).

### ipv4-address *address*

Specifies an IPv4 address.

### ipv6-address *address*

Specifies an IPv6 address.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn l2route type ethernet-segment** command no arguments or keywords are used.

```
device# show bgp evpn l2route type ethernet-segment

Total number of BGP EVPN Ethernet Segment Routes : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
Route Distinguisher: 32.0.0.32:1
1       ESR:[00.000000.000000.00aabb] [IPv4:32.0.0.32]
        33.0.0.33      0        100        0        BL
        AS_PATH:
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route type inclusive-multicast

Displays information about BGP EVPN inclusive multicast routes in the MAC-VRF table.

## Syntax

`show bgp evpn l2route type inclusive-multicast`

`show bgp evpn l2route type inclusive-multicast brief`

`show bgp evpn l2route type inclusive-multicast detail`

`show bgp evpn l2route type inclusive-multicast ethernet-tag tag-id ipv4-address address [ l2-vni number ]`

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**ipv4-address** *address*

Specifies an IPv4 address.

**l2-vni** *number*

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn l2route type inclusive-multicast** command when no arguments or keywords are used.

```
device# show bgp evpn l2route type inclusive-multicast

Total number of BGP EVPN IMR Routes : 1001
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop           MED           LocPrf        Weight Status
Route Distinguisher: 6.0.0.6:1
1      IMR:[21][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 21
2      IMR:[23][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 23
3      IMR:[24][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 24
4      IMR:[25][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 25
5      IMR:[26][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 26
6      IMR:[27][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 27
7      IMR:[28][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 28
8      IMR:[29][IPv4:18.0.0.18]
       18.0.0.18           0             100           0            BE
       AS_PATH: 65002 65012
       L2_vni: 29
...
```

The following example shows output for the **show bgp evpn l2route type inclusive-multicast** command when the **brief** keyword is used.

```
device# show bgp evpn l2route type inclusive-multicast brief

Total number of BGP EVPN IMR Routes : 1001
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network           Next Hop           MED           LocPrf       Weight Path
Route Distinguisher: 6.0.0.6:1
*>  IMR:[21][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
*>  IMR:[23][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
*>  IMR:[24][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
*>  IMR:[25][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
*>  IMR:[26][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
*>  IMR:[27][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
*>  IMR:[28][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
*>  IMR:[29][IPv4:18.0.0.18]
                                18.0.0.18           100            0           65002 65012 ?
...
```

The following example shows output for the **show bgp evpn l2route type inclusive-multicast** command when the **detail** keyword is used.

```
device# show bgp evpn l2route type inclusive-multicast detail

Total number of BGP EVPN IMR Routes : 1001
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:1
1     Prefix: IMR:[21][IPv4:18.0.0.18], Status: BE, Age: 17h27m34s
      NEXT_HOP: 18.0.0.18, Learned from Peer: 5.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65012
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65012:21
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000015 Tunnel-Type: 0x00000006 Tunnel-
IP: 18.0.0.18
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 21
2     Prefix: IMR:[23][IPv4:18.0.0.18], Status: BE, Age: 17h27m34s
      NEXT_HOP: 18.0.0.18, Learned from Peer: 4.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65012
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65012:23
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000017 Tunnel-Type: 0x00000006 Tunnel-
IP: 18.0.0.18
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23
3     Prefix: IMR:[24][IPv4:18.0.0.18], Status: BE, Age: 17h27m34s
      NEXT_HOP: 18.0.0.18, Learned from Peer: 4.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65012
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65012:24
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000018 Tunnel-Type: 0x00000006 Tunnel-
IP: 18.0.0.18
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 24
4     Prefix: IMR:[25][IPv4:18.0.0.18], Status: BE, Age: 17h27m34s
      NEXT_HOP: 18.0.0.18, Learned from Peer: 4.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65012
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65012:25
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000019 Tunnel-Type: 0x00000006 Tunnel-
IP: 18.0.0.18
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 25
5     Prefix: IMR:[26][IPv4:18.0.0.18], Status: BE, Age: 17h27m34s
      NEXT_HOP: 18.0.0.18, Learned from Peer: 4.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65012
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65012:26
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x0000001a Tunnel-Type: 0x00000006 Tunnel-
IP: 18.0.0.18
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 26
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route type mac

Displays information about BGP EVPN Media Access Control (MAC) routes in the MAC-VRF table.

## Syntax

`show bgp evpn l2route type mac`

`show bgp evpn l2route type mac brief`

`show bgp evpn l2route type mac detail`

`show bgp evpn l2route type mac mac address ethernet-tag tag-id [ l2-vni number ]`

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

**mac** *mac address*

Specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**l2-vni**

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

## Modes

Privileged EXEC mode



## Examples

The following example shows output for the **show bgp evpn l2route type mac** command when no arguments or keywords are used.

```
device# show bgp evpn l2route type mac

Total number of BGP EVPN MAC Routes : 2667
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop           MED           LocPrf        Weight Status
Route Distinguisher: 6.0.0.6:1
1      MAC:[23][0000.abba.abba]
      0.0.0.0           0             100           0             BL
      AS_PATH:
      L2_vni: 23
      ESI : 00.00000000000000000000
2      MAC:[23][0000.abba.abba]
      19.0.0.19         0             100           0             E
      AS_PATH: 65002 65110
      L2_vni: 23
      ESI : 00.00000000000000000000
3      MAC:[23][0000.abba.abba]
      76.0.0.76         0             100           0             E
      AS_PATH: 65002 65110
      L2_vni: 23
      ESI : 00.00000000000000000000
4      MAC:[23][0000.abba.abba]
      18.0.0.18         0             100           0             E
      AS_PATH: 65002 65012
      L2_vni: 23
      ESI : 00.00000000000000000000
5      MAC:[23][0000.abba.abba]
      76.0.0.76         0             100           0             E
      AS_PATH: 65002 65110
      L2_vni: 23
      ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn l2route type mac** command when the **brief** keyword is used.

```
device# show bgp evpn l2route type mac brief

Total number of BGP EVPN MAC Routes : 2667
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop           MED           LocPrf        Weight Path
Route Distinguisher: 6.0.0.6:1
*>  MAC:[23][0000.abba.abba]
      0.0.0.0           0             100           0             ?
*   MAC:[23][0000.abba.abba]
      19.0.0.19         100           0             65002 65110 ?
*   MAC:[23][0000.abba.abba]
      76.0.0.76         100           0             65002 65110 ?
*   MAC:[23][0000.abba.abba]
      18.0.0.18         100           0             65002 65012 ?
*   MAC:[23][0000.abba.abba]
      76.0.0.76         100           0             65002 65110 ?
*   MAC:[23][0000.abba.abba]
      23.0.0.23         100           0             65002 65078 ?
      MAC:[23][0000.abba.abba]
      23.0.0.23         100           0             65002 65078 ?
*>  MAC:[23][0000.abba.baba]
      0.0.0.0           0             100           0             ?
*   MAC:[23][0000.abba.baba]
      19.0.0.19         100           0             65002 65110 ?
...
```

The following example shows output for the **show bgp evpn l2route type mac** command when the **detail** keyword is used.

```

device# show bgp evpn l2route type mac detail
Total number of BGP EVPN MAC Routes : 2667
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:1
1      Prefix: MAC:[23][0000.abba.abba], Status: BL, Age: 17h43m46s
      NEXT_HOP: 0.0.0.0, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
      AS_PATH:
      _Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23
      ESI : 00.000000000000000000
2      Prefix: MAC:[23][0000.abba.abba], Status: E, Age: 17h42m16s
      NEXT_HOP: 19.0.0.19, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
      _Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65110:23
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23
      ESI : 00.000000000000000000
3      Prefix: MAC:[23][0000.abba.abba], Status: E, Age: 17h42m29s
      NEXT_HOP: 76.0.0.76, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
      _Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65110:23
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23
      ESI : 00.000000000000000000
4      Prefix: MAC:[23][0000.abba.abba], Status: E, Age: 17h42m42s
      NEXT_HOP: 18.0.0.18, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65012
      _Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65012:23
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23
      ESI : 00.000000000000000000
5      Prefix: MAC:[23][0000.abba.abba], Status: E, Age: 17h42m59s
      NEXT_HOP: 76.0.0.76, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
      _Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65110:23
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23
      ESI : 00.000000000000000000
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route type nd

Displays information about BGP EVPN neighbor-discovery (ND) routes in the MAC-VRF table.

## Syntax

**show bgp evpn l2route type nd**

**show bgp evpn l2route type nd brief**

**show bgp evpn l2route type nd detail**

**show bgp evpn l2route type nd *IPv6 address* mac *mac address* ethernet-tag *tag-id***

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

*IPv6 address*

Specifies an IPv6 address.

**mac** *mac address*

Specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn l2route type nd** command when no arguments or keywords are used.

```
device# show bgp evpn l2route type nd

Total number of BGP EVPN ND Routes : 1778
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop           MED           LocPrf         Weight Status
Route Distinguisher: 6.0.0.6:1
1      ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      0.0.0.0           0             100            0             BL
      AS_PATH:
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
2      ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      19.0.0.19         0             100            0             E
      AS_PATH: 65002 65110
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
3      ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      76.0.0.76         0             100            0             E
      AS_PATH: 65002 65110
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
4      ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      18.0.0.18         0             100            0             E
      AS_PATH: 65002 65012
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
5      ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      76.0.0.76         0             100            0             E
      AS_PATH: 65002 65110
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn l2route type nd** command when the **brief** keyword is used.

```
device# show bgp evpn l2route type nd brief

Total number of BGP EVPN ND Routes : 1778
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop           MED           LocPrf         Weight Path
Route Distinguisher: 6.0.0.6:1
*>  ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      0.0.0.0           0             100            0             ?
*   ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      19.0.0.19         0             100            0             65002 65110 ?
*   ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      76.0.0.76         0             100            0             65002 65110 ?
*   ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      18.0.0.18         0             100            0             65002 65012 ?
*   ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      76.0.0.76         0             100            0             65002 65110 ?
*   ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      23.0.0.23         0             100            0             65002 65078 ?
      ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254]
      23.0.0.23         0             100            0             65002 65078 ?
*>  ARP:[23][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      0.0.0.0           0             100            0             ?
*>  ARP:[23][50eb.1a13.179a]:[IPv6:fe80::52eb:1aff:fe13:179a]
      19.0.0.19         0             100            0             65002 65110 ?
...
```

The following example shows output for the **show bgp evpn l2route type nd** command when the **detail** keyword is used.

```

device# show bgp evpn l2route type nd detail

Total number of BGP EVPN ND Routes : 1778
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:1
1     Prefix: ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254], Status: BL, Age: 17h57m9s
      NEXT_HOP: 0.0.0.0, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
      AS_PATH:
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
        Default Extd Gw Community: Received
        Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
        L2_vni: 23 L3_vni: 0
        ESI : 00.00000000000000000000
2     Prefix: ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254], Status: E, Age: 17h55m39s
      NEXT_HOP: 19.0.0.19, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65110:23
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
3     Prefix: ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254], Status: E, Age: 17h55m52s
      NEXT_HOP: 76.0.0.76, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65110:23
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
4     Prefix: ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254], Status: E, Age: 17h56m5s
      NEXT_HOP: 18.0.0.18, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65012
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65012:23
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
5     Prefix: ARP:[23][0000.abba.abba]:[IPv6:2:23:1::254], Status: E, Age: 17h56m23s
      NEXT_HOP: 76.0.0.76, Learned from Peer: 3.6.0.0 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65110
        Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65110:23
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn l2route unreachable

Displays information about unreachable BGP EVPN routes in the MAC-VRF table.

## Syntax

```
show bgp evpn l2route unreachable
```

```
show bgp evpn l2route unreachable type { arp | auto-discovery | ethernet-segment | inclusive-multicast | mac | nd }
```

## Parameters

### type

Specifies the type of Layer 2 route.

### arp

Specifies Address Resolution Protocol (ARP) routes.

### auto-discovery

Specifies automatically discovered routes.

### ethernet-segment

Specifies Ethernet Segments (ES) routes.

### inclusive-multicast

Specifies inclusive multicast routes.

### mac

Specifies MAC routes.

### nd

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn l2route unreachable** command.

```
device# show bgp evpn l2route unreachable

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix          Next Hop          MED          LocPrf        Weight Status
1  IMR:[22][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 22
2  IMR:[23][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 23
3  IMR:[24][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 24
4  IMR:[25][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 25
5  IMR:[26][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 26
6  IMR:[27][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 27
7  IMR:[28][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 28
8  IMR:[29][IPv4:76.0.0.76]
    76.0.0.76      0            100           0             I
    AS_PATH:
    L2_vni: 29
...
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show bgp evpn l3vni

Displays BGP EVPN information for Layer 3 virtual network identifiers (VNIs).

## Syntax

```
show bgp evpn l3 vni { all-vrfs | vrf name }
```

## Parameters

**all-vrfs**

Specifies all VRFs.

**vrf *name***

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Examples

The following example shows Layer 3 VNI information for all VRFs.

```
device# show bgp evpn l3vni all-vrfs
```

```
-----
L3VNI Prefix Origination Conditions for vrf (2)
-----
Address Family under BGP : True
RD Configured           : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
Router mac Exists       : True
L3 VNI Link UP          : True
Source VTEP             : 0x06000006
L3VNI Active            : Active

-----
L3VNI Prefix Import Conditions for vrf (2)
-----
Address Family under BGP : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
Router mac Exists       : True
L3VNI Active            : Active

-----
L3VNI Prefix Origination Conditions for vrf (3)
-----
Address Family under BGP : True
RD Configured           : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
Router mac Exists       : True
L3 VNI Link UP          : True
Source VTEP             : 0x06000006
L3VNI Active            : Active

-----
L3VNI Prefix Import Conditions for vrf (3)
-----
Address Family under BGP : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
Router mac Exists       : True
L3VNI Active            : Active

-----
L3VNI Prefix Origination Conditions for vrf (4)
-----
Address Family under BGP : True
RD Configured           : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
Router mac Exists       : True
L3 VNI Link UP          : True
Source VTEP             : 0x06000006
L3VNI Active            : Active

-----
L3VNI Prefix Import Conditions for vrf (4)
-----
Address Family under BGP : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
```

```
Router mac Exists      : True
L3VNI Active          : Active
```

-----  
 L3VNI Prefix Origination Conditions for vrf (5)  
 -----

```
Address Family under BGP : True
RD Configured            : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
Router mac Exists       : True
L3 VNI Link UP          : True
Source VTEP              : 0x06000006
L3VNI Active            : Active
```

-----  
 L3VNI Prefix Import Conditions for vrf (5)  
 -----

```
Address Family under BGP : True
L3 VNI Configured       : True
VLAN VNI Mapping exists : True
Router mac Exists       : True
L3VNI Active            : Active
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors

Displays configuration information for BGP EVPN neighbors of the device.

## Syntax

```
show bgp evpn neighbors [ ip-addr | ipv6-addr | routes-summary ]
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

**routes-summary**

Displays routes received, routes accepted, number of routes advertised by peer, and so on.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to view configuration information and statistics for BGP EVPN neighbors of the device. Output shows all configured parameters for the neighbors.

## Examples

The following example shows sample output from the show bgp evpn neighbors command.

```

device# show bgp evpn neighbors

Total number of BGP Neighbors: 3
1  IP Address: 2.0.0.2, AS: 65002 (EBGP), RouterID: 2.0.0.2, VRF: default-vrf
   State: ESTABLISHED, Time: 1d8h24m25s, KeepAliveTime: 60, HoldTime: 180
     KeepAliveTimer Expire in 50 seconds, HoldTimer Expire in 174 seconds
   Minimal Route Advertisement Interval: 0 seconds
     PeerGroup: SpineSwitches
     Multihop-EBGP: yes, ttl: default
     MD5 Password: $TDk1UHNkRClafDg=
     Updatesource: Loopback 1
     RefreshCapability: Received
   Address Family : L2VPN EVPN
     SendExtendedCommunity: yes
   Messages:      Open      Update  KeepAlive Notification Refresh-Req
     Sent       : 1        8062   2171    0         0
     Received: 1        2259   2181    0         0
   Last Update Time: NLRI                               Withdraw      NLRI                               Withdraw
                   Tx: 17h46m0s                        17h45m5s      Rx: 17h50m32s                       17h46m0s
   Last Connection Reset Reason:Unknown
   Notification Sent:      Unspecified
   Notification Received: Unspecified
   Neighbor NLRI Negotiation:
     Peer Negotiated L2VPN EVPN address family
     Peer configured for IPV4 unicast Routes
     Peer configured for L2VPN EVPN address family
   Neighbor ipv6 MPLS Label Capability Negotiation:
   Neighbor AS4 Capability Negotiation:
   Outbound Policy Group:
     ID: 2, Use Count: 1
     Byte Sent: 1641671, Received: 0
     Local host: 9.0.0.9, Local Port: 8127
     Remote host: 2.0.0.2, Remote Port: 179

2  IP Address: 3.0.0.3, AS: 65002 (EBGP), RouterID: 3.0.0.3, VRF: default-vrf
   State: ESTABLISHED, Time: 1d8h24m28s, KeepAliveTime: 60, HoldTime: 180
     KeepAliveTimer Expire in 32 seconds, HoldTimer Expire in 164 seconds
   Minimal Route Advertisement Interval: 0 seconds
     PeerGroup: SpineSwitches
     Multihop-EBGP: yes, ttl: default
     MD5 Password: $TDk1UHNkRClafDg=
     Updatesource: Loopback 1
     RefreshCapability: Received
   Address Family : L2VPN EVPN
     SendExtendedCommunity: yes
   Messages:      Open      Update  KeepAlive Notification Refresh-Req
     Sent       : 1        9155   2176    0         0
     Received: 1        2688   2183    0         0
   Last Update Time: NLRI                               Withdraw      NLRI                               Withdraw
                   Tx: 17h50m36s                        17h46m4s      Rx: 17h46m4s                       17h46m3s
   Last Connection Reset Reason:Unknown
   Notification Sent:      Unspecified
   Notification Received: Unspecified
   Neighbor NLRI Negotiation:
     Peer Negotiated L2VPN EVPN address family
     Peer configured for IPV4 unicast Routes
     Peer configured for L2VPN EVPN address family
   Neighbor ipv6 MPLS Label Capability Negotiation:
   Neighbor AS4 Capability Negotiation:
   Outbound Policy Group:
     ID: 3, Use Count: 1
     Byte Sent: 1833576, Received: 0
     Local host: 9.0.0.9, Local Port: 8021
     Remote host: 3.0.0.3, Remote Port: 179

3  IP Address: 200.1.1.76, AS: 65003 (EBGP), RouterID: 10.0.0.10, VRF: default-vrf
   State: ESTABLISHED, Time: 1d5h58m28s, KeepAliveTime: 60, HoldTime: 180

```

show bgp evpn neighbors

```
KeepAliveTimer Expire in 25 seconds, HoldTimer Expire in 165 seconds
Minimal Route Advertisement Interval: 0 seconds
RefreshCapability: Received
Address Family : L2VPN EVPN
SendExtendedCommunity: yes
Messages:      Open      Update  KeepAlive Notification Refresh-Req
Sent       : 150      4687   2045    147      0
Received: 149      6688   2057    1         0
Last Update Time: NLRI                               Withdraw NLRI                               Withdraw
                  Tx: 17h46m10s                       17h46m9s                               Rx: 17h50m46s                       17h50m44s
Last Connection Reset Reason:Bad Peer AS Number
Notification Sent:      Open Message Error/Bad Peer AS Number
Notification Received: Cease/Administrative Reset
Neighbor NLRI Negotiation:
Peer Negotiated L2VPN EVPN address family
Peer configured for IPV4 unicast Routes
Peer configured for L2VPN EVPN address family
Neighbor ipv6 MPLS Label Capability Negotiation:
Neighbor AS4 Capability Negotiation:
Outbound Policy Group:
ID: 5, Use Count: 1
Byte Sent: 541521, Received: 0
Local host: 200.1.1.109, Local Port: 8109
Remote host: 200.1.1.76, Remote Port: 179
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors advertised-routes detail

Displays detailed information about the routes that the device has advertised to the specified neighbor during the current BGP EVPN session.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } advertised-routes detail
```

```
show bgp evpn neighbors { ip address | ipv6 address } advertised-routes detail type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix | mac | nd }
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

**type**

Specifies the type of route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segment (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**ipv4-prefix**

Specifies IPv4 prefix routes.

**ipv6-prefix**

Specifies IPv6 prefix routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn neighbors advertised-routes detail** command.

```
device# show bgp evpn neighbors 2.0.0.2 advertised-routes detail

There are 5812 routes advertised to neighbor 2.0.0.2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
1 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.29.1.254], Status: BE, Age: 1d6h1m40s
  NEXT_HOP: 19.0.0.19, Learned from Peer: 200.1.1.76 (65003)
  LOCAL_PREF: none, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65003
  Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:00:00:08 RT
65003:29 RT 2:2 RT 65003:20 ExtCom:06:03:50:eb:1a:13:17:9a
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 29 L3_vni: 20 Router Mac : 50:eb:1a:13:17:9a
  ESI : 00.00000000000000000000000000000000
2 Prefix: ND:[0][0000.abba.abba]:[IPv6:2:29:1::254], Status: BE, Age: 1d6h1m40s
  NEXT_HOP: 19.0.0.19, Learned from Peer: 200.1.1.76 (65003)
  LOCAL_PREF: none, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65003
  Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:00:00:08 RT
65003:29 RT 2:2 RT 65003:20 ExtCom:06:03:50:eb:1a:13:17:9a
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 29 L3_vni: 20 Router Mac : 50:eb:1a:13:17:9a
  ESI : 00.00000000000000000000000000000000
3 Prefix: MAC:[0][50eb.1a13.8074], Status: BE, Age: 1d6h1m35s
  NEXT_HOP: 76.0.0.76, Learned from Peer: 200.1.1.76 (65003)
  LOCAL_PREF: none, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65003
  Extended Community: ExtCom:06:00:01:00:00:00:00:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65003:136
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 136
  ESI : 00.00000000000000000000000000000000
4 Prefix: MAC:[0][0000.abba.baba], Status: BE, Age: 1d6h1m35s
  NEXT_HOP: 76.0.0.76, Learned from Peer: 200.1.1.76 (65003)
  LOCAL_PREF: none, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65003
  Extended Community: ExtCom:06:00:01:00:00:00:00:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65003:136
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 136
  ESI : 00.00000000000000000000000000000000
5 Prefix: MAC:[0][0000.abba.abba], Status: BE, Age: 1d6h1m35s
  NEXT_HOP: 76.0.0.76, Learned from Peer: 200.1.1.76 (65003)
  LOCAL_PREF: none, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65003
  Extended Community: ExtCom:06:00:01:00:00:00:00:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65003:136
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 136
  ESI : 00.00000000000000000000000000000000
...
```



## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors advertised-routes type

Displays information about the routes that the device has advertised to the specified neighbor during the current BGP EVPN session.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } advertised-routes type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix | mac | nd }
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

**type**

Specifies the type of route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segments (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**ipv4-prefix**

Specifies IPv4 prefix routes.

**ipv6-prefix**

Specifies IPv6 prefix routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn neighbors advertised-routes type** command when the **arp** keyword is used.

```
device# show bgp evpn neighbors 2.0.0.2 advertised-routes type arp

      There are 5812 routes advertised to neighbor 2.0.0.2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
Prefix      Next Hop      MED      LocPrf      Weight  Status
1  ARP:[0][0000.abba.baba]:[IPv4:2.29.1.254]
    19.0.0.19      0
    AS_PATH: 65009 65003
    L2_vni: 29 L3_vni: 20 Router Mac : 50:eb:1a:13:17:9a
    ESI : 00.00000000000000000000
2  ARP:[0][0000.abba.baba]:[IPv4:9.92.1.254]
    18.0.0.18      0
    AS_PATH: 65009 65003
    L2_vni: 92 L3_vni: 90 Router Mac : 50:eb:1a:14:08:0b
    ESI : 00.00000000000000000000
3  ARP:[0][0000.abba.baba]:[IPv4:9.93.1.254]
    18.0.0.18      0
    AS_PATH: 65009 65003
    L2_vni: 93 L3_vni: 90 Router Mac : 50:eb:1a:14:08:0b
    ESI : 00.00000000000000000000
4  ARP:[0][0000.abba.baba]:[IPv4:9.94.1.254]
    18.0.0.18      0
    AS_PATH: 65009 65003
    L2_vni: 94 L3_vni: 90 Router Mac : 50:eb:1a:14:08:0b
    ESI : 00.00000000000000000000
5  ARP:[0][0000.abba.baba]:[IPv4:9.95.1.254]
    18.0.0.18      0
    AS_PATH: 65009 65003
    L2_vni: 95 L3_vni: 90 Router Mac : 50:eb:1a:14:08:0b
    ESI : 00.00000000000000000000
6  ARP:[0][0000.abba.baba]:[IPv4:8.82.1.254]
    109.0.0.109    0
    AS_PATH: 65009
    L2_vni: 82 L3_vni: 80 Router Mac : 00:27:f8:fd:27:4b
    ESI : 00.00000000000000000000
...
```

show bgp evpn neighbors advertised-routes type

The following example shows output for the **show bgp evpn neighbors advertised-routes type** command when the **mac** keyword is used.

```
device# show bgp evpn neighbors 2.0.0.2 advertised-routes type mac

      There are 5812 routes advertised to neighbor 2.0.0.2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
Prefix      Next Hop      MED      LocPrf      Weight  Status
1  MAC:[0][50eb.1a13.8074]
      76.0.0.76      0
      AS_PATH: 65009 65003
      L2_vni: 136
      ESI : 00.00000000000000000000
2  MAC:[0][0000.abba.baba]
      76.0.0.76      0
      AS_PATH: 65009 65003
      L2_vni: 136
      ESI : 00.00000000000000000000
3  MAC:[0][0000.abba.abba]
      76.0.0.76      0
      AS_PATH: 65009 65003
      L2_vni: 136
      ESI : 00.00000000000000000000
4  MAC:[0][0000.abba.abba]
      76.0.0.76      0
      AS_PATH: 65009 65003
      L2_vni: 136
      ESI : 00.00000000000000000000
5  MAC:[0][0000.abba.baba]
      76.0.0.76      0
      AS_PATH: 65009 65003
      L2_vni: 136
      ESI : 00.00000000000000000000
6  MAC:[0][50eb.1a13.df29]
      76.0.0.76      0
      AS_PATH: 65009 65003
      L2_vni: 136
      ESI : 00.00000000000000000000
7  MAC:[0][50eb.1a13.8074]
      76.0.0.76      0
      AS_PATH: 65009 65003
      L2_vni: 137
      ESI : 00.00000000000000000000
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors routes best

Displays routes received from specified BGP EVPN neighbors that are the best BGP EVPN routes to their destination.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } routes best
```

```
show bgp evpn neighbors { ip address | ipv6 address } routes best type { arp | auto-discovery | ethernet-segment | inclusive-multicast | mac | nd }
```

## Parameters

### type

Specifies the type of route.

### arp

Specifies Address Resolution Protocol (ARP) routes.

### auto-discovery

Specifies automatically discovered routes.

### ethernet-segment

Specifies Ethernet Segments (ES) routes.

### inclusive-multicast

Specifies inclusive multicast routes.

### mac

Specifies MAC routes.

### nd

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn neighbors routes best** command.

```
device# show bgp evpn neighbors 2.0.0.2 routes best

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
1  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22
2  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
3  ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
4  ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
5  MAC:[0][0000.abba.abba]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn neighbors routes best** command when the **nd** keyword is used.

```
device# show bgp evpn neighbors 2.0.0.2 routes best type nd

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
1  ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
2  ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
3  ND:[0][0000.abba.abba]:[IPv6:2:23:1::254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
4  ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
5  ND:[0][0000.abba.abba]:[IPv6:2:24:1::254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 24 L3_vni: 0
      ESI : 00.00000000000000000000
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors routes detail

Lists in detail a variety of route information received in messages from a specified BGP EVPN neighbor.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } routes detail
```

```
show bgp evpn neighbors { ip address | ipv6 address } routes detail type { arp | auto-discovery | ethernet-segment |  
inclusive-multicast | mac | nd }
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

**type**

Specifies the type of route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segments (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode



## Examples

The following example shows output for the **show bgp evpn neighbors routes detail** command.

```
device# show bgp evpn neighbors 2.0.0.2 routes detail

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: IMR:[0][IPv4:57.0.0.57], Status: BE, Age: 1d8h31m15s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:22
  PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000016 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
  Extended Community: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22
2 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254], Status: BE, Age: 1d8h30m47s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22 L3_vni: 0
  ESI : 00.000000000000000000
3 Prefix: ND:[0][0000.abba.baba]:[IPv6:2:22:1::254], Status: BE, Age: 1d8h30m47s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22 L3_vni: 0
  ESI : 00.000000000000000000
4 Prefix: ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba], Status: BE, Age: 1d8h30m44s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
  Mac Mobility Sticky: True
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22 L3_vni: 0
  ESI : 00.000000000000000000
5 Prefix: MAC:[0][0000.abba.abba], Status: BE, Age: 1d8h30m47s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22
  ESI : 00.000000000000000000
6 Prefix: MAC:[0][0000.abba.baba], Status: BE, Age: 1d8h30m47s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
  Mac Mobility Sticky: True
```

show bgp evpn neighbors routes detail

```
Default Extd Gw Community: Received
Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
Adj_RIB_out count: 2, Admin distance 20
L2_vni: 22
ESI : 00.000000000000000000
```

...

The following example shows output for the **show bgp evpn neighbors routes detail** command when the **auto-discovery** keyword is used.

```
device# show bgp evpn neighbors 2.0.0.2 routes detail type auto-discovery

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: AD:[01.006501e05200000a00][4294967295], Status: BME, Age: 1d5h2m2s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
    Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
    ESI Label Ext Community: 0 All-Active
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 2, Admin distance 20
2 Prefix: AD:[01.006501e05200000a00][4294967295], Status: BME, Age: 1d5h2m2s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
    Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
    ESI Label Ext Community: 0 All-Active
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 2, Admin distance 20
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors routes not-installed-best

Displays routes received from a BGP EVPN neighbor that are the best routes to their destination but were not installed in the route table due to various limitations, such as hardware entries being already exhausted.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } routes not-installed-best
```

```
show bgp evpn neighbors { ip address | ipv6 address } routes not-installed-best type { arp | auto-discovery | ethernet-  
segment | inclusive-multicast | mac | nd }
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

**type**

Specifies the type of route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segments (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors routes type

Lists a variety of route information received in messages from a specified BGP EVPN neighbor.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } routes type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix | mac | nd }
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

**type**

Specifies the type of route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segments (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**ipv4-prefix**

Specifies IPv4 prefix routes.

**ipv6-prefix**

Specifies IPv6 prefix routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn neighbors routes type** command when the **arp** keyword is used.

```
device# show bgp evpn neighbors 2.0.0.2 routes type arp

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight      Status
1  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 22 L3_vni: 0
    ESI : 00.00000000000000000000
2  ARP:[0][0000.abba.baba]:[IPv4:2.23.1.254]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 23 L3_vni: 0
    ESI : 00.00000000000000000000
3  ARP:[0][0000.abba.baba]:[IPv4:2.24.1.254]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 24 L3_vni: 0
    ESI : 00.00000000000000000000
4  ARP:[0][0000.abba.baba]:[IPv4:2.25.1.254]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 25 L3_vni: 0
    ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn neighbors routes type** command when the **mac** keyword is used.

```
device# show bgp evpn neighbors 2.0.0.2 routes type mac

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight      Status
1  MAC:[0][0000.abba.abba]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 22
    ESI : 00.00000000000000000000
2  MAC:[0][0000.abba.baba]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 22
    ESI : 00.00000000000000000000
3  MAC:[0][0027.f8ca.76ba]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 22
    ESI : 00.00000000000000000000
4  MAC:[0][0000.abba.abba]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 23
    ESI : 00.00000000000000000000
5  MAC:[0][0000.abba.baba]
    57.0.0.57      0      100      0      BE
    AS_PATH: 65002 65006
    L2_vni: 23
    ESI : 00.00000000000000000000
...
```

show bgp evpn neighbors routes type

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn neighbors routes unreachable

Displays BGP EVPN neighbor route information about routes whose destinations are unreachable through any of the BGP EVPN paths in the BGP route table.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } routes unreachable
```

```
show bgp evpn neighbors { ip address | ipv6 address } routes unreachable type { arp | auto-discovery | ethernet-segment | inclusive-multicast | mac | nd }
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

**type**

Specifies the type of route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segments (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn neighbors routes unreachable** command.

```
device# show bgp evpn neighbors 5.0.0.5 routes unreachable

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix          Next Hop          MED          LocPrf      Weight  Status
1  IP4Prefix:[0][2.22.1.0/24]
    76.0.0.76      0            100         0          I
    AS_PATH:
    L3_vni: 20 Router Mac : 50:eb:1a:13:df:29
2  IP4Prefix:[0][2.23.1.0/24]
    76.0.0.76      0            100         0          I
    AS_PATH:
    L3_vni: 20 Router Mac : 50:eb:1a:13:df:29
3  IP4Prefix:[0][2.24.1.0/24]
    76.0.0.76      0            100         0          I
    AS_PATH:
    L3_vni: 20 Router Mac : 50:eb:1a:13:df:29
4  IP4Prefix:[0][2.25.1.0/24]
    76.0.0.76      0            100         0          I
    AS_PATH:
    L3_vni: 20 Router Mac : 50:eb:1a:13:df:29
5  IP4Prefix:[0][2.26.1.0/24]
    76.0.0.76      0            100         0          I
    AS_PATH:
    L3_vni: 20 Router Mac : 50:eb:1a:13:df:29
6  IP4Prefix:[0][2.27.1.0/24]
    76.0.0.76      0            100         0          I
    AS_PATH:
    L3_vni: 20 Router Mac : 50:eb:1a:13:df:29
7  IP4Prefix:[0][2.28.1.0/24]
    76.0.0.76      0            100         0          I
    AS_PATH:
    L3_vni: 20 Router Mac : 50:eb:1a:13:df:29
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show bgp evpn neighbors routes-summary

Lists summarized route information received in messages from specified BGP EVPN neighbors.

## Syntax

```
show bgp evpn neighbors { ip address | ipv6 address } routes-summary
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn neighbors routes-summary** command.

```
device# show bgp evpn neighbors 2.0.0.2 routes-summary

1 IP Address: 2.0.0.2
Routes Accepted/Installed:3202, Filtered/Kept:0, Filtered:2
  Routes Selected as BEST Routes:4067
    BEST Routes not Installed in IP Forwarding Table:0
  Unreachable Routes (no IGP Route for NEXTHOP):0
  History Routes:0

NLRIs Received in Update Message:4453, Withdraws:781 (0), Replacements:0
  NLRIs Discarded due to
    Maximum Prefix Limit:0, AS Loop:0
    Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
    Invalid Confed aspath:0, maxas-limit aspath:0
    Duplicated Originator_ID:0, Cluster_ID:0

Routes Advertised:5812, To be Sent:0, To be Withdrawn:0
NLRIs Sent in Update Message:15080, Withdraws:6893, Replacements:2375

Peer Out of Memory Count for:
  Receiving Update Messages:0, Accepting Routes(NLRI):0
  Attributes:0, Outbound Routes(RIB-out):0 Outbound Routes Holder:0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes

Displays BGP EVPN route information.

## Syntax

**show bgp evpn routes**

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes** command.

```
device# show bgp evpn routes

Total number of BGP EVPN Routes : 12136
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop          MED          LocPrf      Weight Path
Route Distinguisher: 6.0.0.6:32790
*>  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100          0          65002 65006 ?
*   IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100          0          65002 65006 ?
*>  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57          100          0          65002 65006 ?
*   ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57          100          0          65002 65006 ?
*>  ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57          100          0          65002 65006 ?
*   ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57          100          0          65002 65006 ?
*>  ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57          100          0          65002 65006 ?
*   ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57          100          0          65002 65006 ?
*>  MAC:[0][0000.abba.abba]
      57.0.0.57          100          0          65002 65006 ?
*   MAC:[0][0000.abba.abba]
      57.0.0.57          100          0          65002 65006 ?
*>  MAC:[0][0000.abba.baba]
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes best

Displays information for BGP EVPN routes that were selected as best routes.

## Syntax

```
show bgp evpn routes best
```

```
show bgp evpn routes best type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix | mac | nd }
```

## Parameters

### type

Specifies the type of route.

### arp

Specifies Address Resolution Protocol (ARP) routes.

### auto-discovery

Specifies automatically discovered routes.

### ethernet-segment

Specifies Ethernet Segments (ES) routes.

### inclusive-multicast

Specifies inclusive multicast routes.

### ipv4-prefix

Specifies IPv4 prefix routes.

### ipv6-prefix

Specifies IPv6 prefix routes.

### mac

Specifies MAC routes.

### nd

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes best** command.

```
device# show bgp evpn routes best

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
1  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22
2  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
3  ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
4  ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
5  MAC:[0][0000.abba.abba]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
6  MAC:[0][0000.abba.baba]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
...
```

The following example shows output for the **show bgp evpn routes best** command when the **type** and **arp** keywords are used.

```
device# show bgp evpn routes best type arp

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
1  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
2  ARP:[0][0000.abba.baba]:[IPv4:2.23.1.254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
3  ARP:[0][0000.abba.baba]:[IPv4:2.24.1.254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 24 L3_vni: 0
      ESI : 00.00000000000000000000
4  ARP:[0][0000.abba.baba]:[IPv4:2.25.1.254]
      57.0.0.57      0      100      0      BE
      AS_PATH: 65002 65006
      L2_vni: 25 L3_vni: 0
      ESI : 00.00000000000000000000
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

show bgp evpn routes detail

# show bgp evpn routes detail

Displays detailed BGP EVPN route information.

## Syntax

show bgp evpn routes detail

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes detail** command.

```

device# show bgp evpn routes detail
Total number of BGP EVPN Routes : 12136
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:32790
1     Prefix: IMR:[0][IPv4:57.0.0.57], Status: BE, Age: 1d9h20m9s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:22
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000016 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22
2     Prefix: IMR:[0][IPv4:57.0.0.57], Status: E, Age: 1d9h20m9s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:22
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000016 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 22
3     Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254], Status: BE, Age: 1d9h19m42s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:08 RT
65006:22
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
4     Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254], Status: E, Age: 1d9h19m42s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:08 RT
65006:22
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
5     Prefix: ND:[0][0000.abba.abba]:[IPv6:2:22:1::254], Status: BE, Age: 1d9h19m42s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:08 RT
65006:22
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes local

Displays information about BGP EVPN local routes.

## Syntax

```
show bgp evpn routes local
```

```
show bgp evpn routes local type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix | mac | nd }
```

## Parameters

### type

Specifies the type of route.

### arp

Specifies Address Resolution Protocol (ARP) routes.

### auto-discovery

Specifies automatically discovered routes.

### ethernet-segment

Specifies Ethernet Segments (ES) routes.

### inclusive-multicast

Specifies inclusive multicast routes.

### ipv4-prefix

Specifies IPv4 prefix routes.

### ipv6-prefix

Specifies IPv6 prefix routes.

### mac

Specifies MAC routes.

### nd

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode



## Examples

The following example shows output for the **show bgp evpn routes local** command.

```

device# show bgp evpn routes local

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: AD:[00.000000000000989900][4294967295], Status: BL, Age:
1d23h
50m9s
    NEXT_HOP: 78.0.0.78, Learned from Peer: Local Router
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH:
    Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:
00:0
0:00:00:00:08
    ESI Label Ext Community: 0 All-Active
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 2, Admin distance 0
2 Prefix: ESR:[00.000000.000000.989900][IPv4:7.0.0.7], Status: BL,
Age:
1d23h50m9s
    NEXT_HOP: 78.0.0.78, Learned from Peer: Local Router
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH:
    Extended Community: ExtCom:06:02:00:00:00:00:98:99:00 ExtCom:03:0c:
00:0
0:00:00:00:08
    RT Import 0:10000640
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 2, Admin distance 0
3 Prefix: IP4Prefix:[0][21.1.1.0/24], Status: BL, Age: 1d23h36m46s
    NEXT_HOP: 78.0.0.78, Learned from Peer: Local Router
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH:
    Extended Community: RT 1:1 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:
03:
00:00:00:00:00:08
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 2, Admin distance 0
    L3_vni: 10020 Router Mac : 50:eb:1a:14:07:67
4 Prefix: IP4Prefix:[0][22.1.1.0/24], Status: BL, Age: 1d23h36m46s
    NEXT_HOP: 78.0.0.78, Learned from Peer: Local Router
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH:
    Extended Community: RT 1:1 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:
03:
00:00:00:00:00:08
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 2, Admin distance 0
    L3_vni: 10020 Router Mac : 50:eb:1a:14:07:67
...

```

This example shows output for the **show bgp evpn routes local** command when the **type** and **mac** keywords are used.

```

device# show bgp evpn routes local type mac

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: MAC:[0][0000.abba.abba], Status: BL, Age: 1d9h36m12s
  NEXT_HOP: 109.0.0.109, Learned from Peer: Local Router
  LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
  AS_PATH:
    Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65009:22
    Mac Mobility Sticky: True
    Default Extd Gw Community: Received
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 3, Admin distance 0
    L2_vni: 22
    ESI : 00.00000000000000000000
2 Prefix: MAC:[0][0000.abba.baba], Status: BL, Age: 1d9h36m12s
  NEXT_HOP: 109.0.0.109, Learned from Peer: Local Router
  LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
  AS_PATH:
    Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65009:22
    Mac Mobility Sticky: True
    Default Extd Gw Community: Received
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 3, Admin distance 0
    L2_vni: 22
    ESI : 00.00000000000000000000
3 Prefix: MAC:[0][0027.f8fd.274b], Status: BL, Age: 1d9h36m32s
  NEXT_HOP: 109.0.0.109, Learned from Peer: Local Router
  LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
  AS_PATH:
    Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65009:22
    Mac Mobility Sticky: True
    Default Extd Gw Community: Received
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    Adj_RIB_out count: 3, Admin distance 0
    L2_vni: 22
    ESI : 00.00000000000000000000
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes next-hop

Displays information for BGP EVPN routes received from the specified next-hop.

## Syntax

```
show bgp evpn routes next-hop { ipv4-address | ipv4-address } type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix | mac | nd }
```

## Parameters

*ipv4-address*

Specifies an IPv4 address.

*ipv6-address*

Specifies an IPv6 address.

**type**

Specifies the type of route.

**arp**

Specifies Address Resolution Protocol (ARP) routes.

**auto-discovery**

Specifies automatically discovered routes.

**ethernet-segment**

Specifies Ethernet Segments (ES) routes.

**inclusive-multicast**

Specifies inclusive multicast routes.

**ipv4-prefix**

Specifies IPv4 prefix routes.

**ipv6-prefix**

Specifies IPv6 prefix routes.

**mac**

Specifies MAC routes.

**nd**

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes next-hop** command when the **type** and **arp** keywords are used.

```
device# show bgp evpn routes next-hop 57.0.0.57 type arp

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254], Status: BE, Age: 1d8h44m6s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 200.1.1.109 (65009)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:08 RT
65006:22
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22 L3_vni: 0
  ESI : 00.00000000000000000000
2 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.23.1.254], Status: BE, Age: 1d8h44m6s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 200.1.1.109 (65009)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:08 RT
65006:23
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 23 L3_vni: 0
  ESI : 00.00000000000000000000
3 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.24.1.254], Status: BE, Age: 1d8h44m6s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 200.1.1.109 (65009)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:08 RT
65006:24
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 24 L3_vni: 0
  ESI : 00.00000000000000000000
4 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.25.1.254], Status: BE, Age: 1d8h44m6s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 200.1.1.109 (65009)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:08 RT
65006:25
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 25 L3_vni: 0
  ESI : 00.00000000000000000000
5 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.26.1.254], Status: BE, Age: 1d8h44m6s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 200.1.1.109 (65009)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:08 RT
65006:26
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 26 L3_vni: 0
  ESI : 00.00000000000000000000
6 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.27.1.254], Status: BE, Age: 1d8h44m6s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 200.1.1.109 (65009)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65009 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:08 RT
```

```
65006:27
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 27 L3_vni: 0
  ESI : 00.000000000000000000
```

...

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes no-best

Displays information for BGP EVPN routes that were selected as not best routes.

## Syntax

```
show bgp evpn routes no-best
```

```
show bgp evpn routes no-best type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix  
| mac | nd }
```

## Parameters

### type

Specifies the type of route.

### arp

Specifies Address Resolution Protocol (ARP) routes.

### auto-discovery

Specifies automatically discovered routes.

### ethernet-segment

Specifies Ethernet Segments (ES) routes.

### inclusive-multicast

Specifies inclusive multicast routes.

### ipv4-prefix

Specifies IPv4 prefix routes.

### ipv6-prefix

Specifies IPv6 prefix routes.

### mac

Specifies MAC routes.

### nd

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes no-best** command.

```
device# show bgp evpn routes no-best

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
1  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57      0      100      0      I
      AS_PATH: 65009 65002 65006
      L2_vni: 22
2  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57      0      100      0      I
      AS_PATH: 65009 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
3  ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57      0      100      0      I
      AS_PATH: 65009 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
4  ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57      0      100      0      I
      AS_PATH: 65009 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
5  MAC:[0][0000.abba.abba]
      57.0.0.57      0      100      0      I
      AS_PATH: 65009 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
6  MAC:[0][0000.abba.baba]
      57.0.0.57      0      100      0      I
      AS_PATH: 65009 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes not-installed-best

Displays information for BGP EVPN best routes that are not installed.

## Syntax

```
show bgp evpn routes not-installed-best
```

```
show bgp evpn routes not-installed-best type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix |  
        ipv6-prefix | mac | nd }
```

## Parameters

### type

Specifies the type of route.

### arp

Specifies Address Resolution Protocol (ARP) routes.

### auto-discovery

Specifies automatically discovered routes.

### ethernet-segment

Specifies Ethernet Segments (ES) routes.

### inclusive-multicast

Specifies inclusive multicast routes.

### ipv4-prefix

Specifies IPv4 prefix routes.

### ipv6-prefix

Specifies IPv6 prefix routes.

### mac

Specifies MAC routes.

### nd

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## History

Release version	Command history
7.0.0	This command was introduced.



# show bgp evpn routes rd

Displays information for BGP EVPN routes with the specified route distinguisher (RD).

## Syntax

```
show bgp evpn routes rd admin-value:arbitrary-value
```

## Parameters

### *admin-value*

The administrative number assigned to the route. This can be a local ASN number or an IP address. The ASN number can be either a 2-byte number (from 0 through 65535) or a 4-byte number (from 0 through 4294967295).

### *arbitrary-value*

An arbitrary number you choose. The range of valid values is from 0 through 65535 if the ASN is 2 byte, or from 0 through 4294967295 if the ASN is 4 byte.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes rd** command.

```
device# show bgp evpn routes rd 6.0.0.6:32790

Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network      Next Hop      MED      LocPrf      Weight Path
*>  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57
      100      0      65002 65006 ?
*   IMR:[0][IPv4:57.0.0.57]
      57.0.0.57
      100      0      65002 65006 ?
*>  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57
      100      0      65002 65006 ?
*   ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57
      100      0      65002 65006 ?
*>  ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57
      100      0      65002 65006 ?
*   ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
      57.0.0.57
      100      0      65002 65006 ?
*>  ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57
      100      0      65002 65006 ?
*   ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
      57.0.0.57
      100      0      65002 65006 ?
*>  MAC:[0][0000.abba.abba]
      57.0.0.57
      100      0      65002 65006 ?
*   MAC:[0][0000.abba.abba]
      57.0.0.57
      100      0      65002 65006 ?
*>  MAC:[0][0000.abba.baba]
      57.0.0.57
      100      0      65002 65006 ?
*   MAC:[0][0000.abba.baba]
      57.0.0.57
      100      0      65002 65006 ?
*>  MAC:[0][0027.f8ca.76ba]
      57.0.0.57
      100      0      65002 65006 ?
*   MAC:[0][0027.f8ca.76ba]
      57.0.0.57
      100      0      65002 65006 ?
```

show bgp evpn routes rd

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes rd type

Displays information for BGP EVPN routes, filtered based on a specified route type, with the specified route distinguisher (RD).

## Syntax

```
show bgp evpn routes rd admin-value:arbitrary-value type { arp | auto-discovery | ethernet-segment | inclusive-multicast |
  ipv4-prefix | ipv6-prefix | mac | nd } detail

show bgp evpn routes rd admin-value:arbitrary-value type arp ip address mac mac address ethernet-tag tag-id l2vni
  number

show bgp evpn routes rd admin-value:arbitrary-value type auto-discovery esi-value value ethernet-tag tag-id

show bgp evpn routes rd admin-value:arbitrary-value type ethernet-segment esi-value value { ipv4-address address | ipv6-
  address address }

show bgp evpn routes rd admin-value:arbitrary-value type inclusive-multicast ethernet-tag tag-id ipv4-address address
  [ l2-vni number

show bgp evpn routes rd admin-value:arbitrary-value type ipv4-prefix ip address/mask tag tag-id [ l3vni value ]

show bgp evpn routes rd admin-value:arbitrary-value type ipv6-prefix ipv6 address/mask tag tag-id [ l3vni value ]

show bgp evpn routes rd admin-value:arbitrary-value type mac mac address ethernet-tag tag-id [ l2-vni number ]

show bgp evpn routes rd admin-value:arbitrary-value type nd IPv6 address mac mac address ethernet-tag tag-id
```

## Parameters

### *admin-value*

The administrative number assigned to the route. This can be a local ASN number or an IP address. The ASN number can be either a 2-byte number (from 0 through 65535) or a 4-byte number (from 0 through 4294967295).

### *arbitrary-value*

An arbitrary number you choose. The range of valid values is from 0 through 65535 if the ASN is 2 byte, or from 0 through 4294967295 if the ASN is 4 byte.

### **type**

Specifies a route type.

### **arp**

Specifies address-resolution protocol (ARP).

### **auto-discovery**

Specifies automatically discovered routes.

### **ethernet-segment**

Specifies Ethernet Segment (ES) information.

### **inclusive-multicast**

Specifies inclusive multicast information.

### **ipv4-prefix**

Specifies IPv4 prefix information information.

**ipv6-prefix**

Specifies IPv6 prefix information information.

**mac**

Specifies Media Access Control (MAC) information.

**nd**

Specifies neighbor-discovery (ND) information.

**detail**

Displays detailed information.

**mac** *mac address*

Specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**l2-vni** *number*

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

**esi-value** *value*

Specifies a 10 byte Ethernet Segment Identifier (ESI) value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH.HH).

**ipv4-address** *address*

Specifies an IPv4 address.

**ipv6-address** *address*

Specifies an IPv6 address.

*ip address/mask*

Specifies an IPv4 address and mask.

*ipv6 address/mask*

Specifies an IPv6 address and mask.

**tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**l3vni** *value*

Specifies a Layer 3 virtual network identifier (VNIs). Valid values range from 1 through 6777215.

## Modes

Privileged EXEC mode

## Examples

The following example shows detailed ARP information for a BGP EVPN route with the RD 6.0.0.6:32790.

```
device# show bgp evpn routes rd 6.0.0.6:32790 type arp detail

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254], Status: BE, Age: 1d7h23m14s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22 L3_vni: 0
  ESI : 00.00000000000000000000
2 Prefix: ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254], Status: E, Age: 1d7h23m15s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  L2_vni: 22 L3_vni: 0
  ESI : 00.00000000000000000000
```

The following example shows detailed MAC information for a BGP EVPN route with the RD 6.0.0.6:32790.

```

device# show bgp evpn routes rd 6.0.0.6:32790 type mac detail

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: MAC:[0][0000.abba.abba], Status: BE, Age: 1d7h26m34s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22
  ESI : 00.000000000000000000
2 Prefix: MAC:[0][0000.abba.abba], Status: E, Age: 1d7h26m34s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  L2_vni: 22
  ESI : 00.000000000000000000
3 Prefix: MAC:[0][0000.abba.baba], Status: BE, Age: 1d7h26m34s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22
  ESI : 00.000000000000000000
4 Prefix: MAC:[0][0000.abba.baba], Status: E, Age: 1d7h26m34s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  L2_vni: 22
  ESI : 00.000000000000000000
5 Prefix: MAC:[0][0027.f8ca.76ba], Status: BE, Age: 1d7h27m2s
  NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65006
  Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
  Mac Mobility Sticky: True
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L2_vni: 22
  ESI : 00.000000000000000000
...

```

The following example shows detailed inclusive multicast information for a BGP EVPN route with the RD 6.0.0.6:32790.

```
device# show bgp evpn routes rd 6.0.0.6:32790 type inclusive-multicast detail

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
1      Prefix: IMR:[0][IPv4:57.0.0.57], Status: BE, Age: 1d7h34m37s
       NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
       LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
       AS_PATH: 65002 65006
       Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:22
       PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000016 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
       Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
       Adj_RIB_out count: 2, Admin distance 20
       L2_vni: 22
2      Prefix: IMR:[0][IPv4:57.0.0.57], Status: E, Age: 1d7h34m37s
       NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
       LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
       AS_PATH: 65002 65006
       Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:22
       PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000016 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
       Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
       L2_vni: 22
```

## History

Release version	Command history
7.0.0	This command was introduced.

show bgp evpn routes summary

# show bgp evpn routes summary

Displays BGP EVPN summary route information.

## Syntax

`show bgp evpn routes summary`

## Modes

Privileged EXEC mode



## Examples

The following example shows summarized route information for BGP EVPN.

```
device# show bgp evpn routes summary

Total number of BGP EVPN routes (NLRIs) Installed      : 12136
Distinct BGP destination EVPN Routes                  : 12136
EVPN Routes originated by this router                  : 1152
EVPN Routes selected as BEST routes                   : 8934
EVPN Routes Installed as BEST routes                   : 8934
EVPN BEST routes not installed in IP forwarding table  : 3202
Unreachable EVPN routes (no IGP route for NEXTHOP)    : 0
IBGP EVPN routes selected as best routes               : 0
EBGP EVPN routes selected as best routes               : 7782
BEST EVPN routes not valid for IP forwarding table     : 0

Distinct BGP destination IMR routes                    : 1408

Filtered IMR routes                                    : 0
IMR routes originated by this router                   : 128
IMR routes selected as BEST routes                     : 1024
IMR routes Installed as BEST routes                     : 1024
IMR BEST routes not installed in IP forwarding table   : 384
Unreachable IMR routes (no IGP route for NEXTHOP)    : 0
IBGP IMR routes selected as best routes                 : 0
EBGP IMR routes selected as best routes                 : 896
IMR BEST routes not valid for IP forwarding table     : 0

Distinct BGP destination ARP routes                    : 1404

Filtered ARP routes                                    : 0
ARP routes originated by this router                   : 128
ARP routes selected as BEST routes                     : 1020
ARP routes Installed as BEST routes                     : 1020
ARP BEST routes not installed in IP forwarding table   : 384
Unreachable ARP routes (no IGP route for NEXTHOP)    : 0
IBGP ARP routes selected as best routes                 : 0
EBGP ARP routes selected as best routes                 : 892
ARP BEST routes not valid for IP forwarding table     : 0

Distinct BGP destination ND routes                     : 2808

Filtered ND routes                                    : 0
ND routes originated by this router                     : 256
ND routes selected as BEST routes                       : 2040
ND routes Installed as BEST routes                       : 2040
ND BEST routes not installed in IP forwarding table    : 768
Unreachable ND routes (no IGP route for NEXTHOP)     : 0
IBGP ND routes selected as best routes                   : 0
EBGP ND routes selected as best routes                   : 1784
ND BEST routes not valid for IP forwarding table       : 0

Distinct BGP destination MAC routes                    : 4216

Filtered MAC routes                                    : 0
MAC routes originated by this router                     : 384
MAC routes selected as BEST routes                       : 3064
MAC routes Installed as BEST routes                       : 3064
MAC BEST routes not installed in IP forwarding table    : 1152
Unreachable MAC routes (no IGP route for NEXTHOP)     : 0
IBGP MAC routes selected as best routes                   : 0
EBGP MAC routes selected as best routes                   : 2680
MAC BEST routes not valid for IP forwarding table       : 0

Distinct BGP destination AD routes                     : 4

Filtered AD routes                                    : 0
AD routes originated by this router                       : 0
AD routes selected as BEST routes                         : 2
AD routes Installed as BEST routes                         : 2
```

```

AD BEST routes not installed in IP forwarding table : 2
Unreachable AD routes (no IGP route for NEXTHOP) : 0
IBGP AD routes selected as best routes : 0
EBGP AD routes selected as best routes : 2
AD BEST routes not valid for IP forwarding table : 0

Distinct BGP destination Ipv4_Prefix routes : 1148

Filtered Ipv4_Prefix routes : 0
Ipv4_Prefix routes originated by this router : 128
Ipv4_Prefix routes selected as BEST routes : 892
Ipv4_Prefix routes Installed as BEST routes : 892
Ipv4_Prefix BEST routes not installed in IP forwarding table : 256
Unreachable Ipv4_Prefix routes (no IGP route for NEXTHOP) : 0
IBGP Ipv4_Prefix routes selected as best routes : 0
EBGP Ipv4_Prefix routes selected as best routes : 764
Ipv4_Prefix BEST routes not valid for IP forwarding table : 0

Distinct BGP destination IPV6_Prefix routes : 1148

Filtered IPV6_Prefix routes : 0
IPV6_Prefix routes originated by this router : 128
IPV6_Prefix routes selected as BEST routes : 892
IPV6_Prefix routes Installed as BEST routes : 892
IPV6_Prefix BEST routes not installed in IP forwarding table : 256
Unreachable IPV6_Prefix routes (no IGP route for NEXTHOP) : 0
IBGP IPV6_Prefix routes selected as best routes : 0
EBGP IPV6_Prefix routes selected as best routes : 764
IPV6_Prefix BEST routes not valid for IP forwarding table : 0
    
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes type arp

Displays information for BGP EVPN address-resolution protocol (ARP) routes in the BGP EVPN route table.

## Syntax

```
show bgp evpn routes type arp
```

```
show bgp evpn routes type arp brief
```

```
show bgp evpn routes type arp detail
```

```
show bgp evpn routes type arp ip address mac mac address ethernet-tag tag-id l2vni number
```

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

*ip address*

Specifies the IP address.

**mac** *mac address*

Specifies a Media Access Control (MAC) address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**l2-vni** *number*

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

## Modes

Privileged EXEC mode

## Examples

The following example shows summarized output for the **show bgp evpn routes type arp** command.

```
device# show bgp evpn routes type arp brief

Total number of BGP EVPN ARP Routes : 1404
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop          MED          LocPrf          Weight Path
Route Distinguisher: 6.0.0.6:32790
*>  ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57          100          0          65002 65006 ?
*   ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57          100          0          65002 65006 ?
Route Distinguisher: 6.0.0.6:32791
*>  ARP:[0][0000.abba.baba]:[IPv4:2.23.1.254]
      57.0.0.57          100          0          65002 65006 ?
*   ARP:[0][0000.abba.baba]:[IPv4:2.23.1.254]
      57.0.0.57          100          0          65002 65006 ?
Route Distinguisher: 6.0.0.6:32792
...
```

The following example shows sample output for the **show bgp evpn routes type arp** command.

```
device# show bgp evpn routes type arp

Total number of BGP EVPN ARP Routes : 1404
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
   Prefix          Next Hop          MED          LocPrf          Weight Status
Route Distinguisher: 6.0.0.6:32790
1   ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57          0          100          0          BE
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
2   ARP:[0][0000.abba.baba]:[IPv4:2.22.1.254]
      57.0.0.57          0          100          0          E
      AS_PATH: 65002 65006
      L2_vni: 22 L3_vni: 0
      ESI : 00.00000000000000000000
Route Distinguisher: 6.0.0.6:32791
3   ARP:[0][0000.abba.baba]:[IPv4:2.23.1.254]
      57.0.0.57          0          100          0          BE
      AS_PATH: 65002 65006
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
4   ARP:[0][0000.abba.baba]:[IPv4:2.23.1.254]
      57.0.0.57          0          100          0          E
      AS_PATH: 65002 65006
      L2_vni: 23 L3_vni: 0
      ESI : 00.00000000000000000000
Route Distinguisher: 6.0.0.6:32792
5   ARP:[0][0000.abba.baba]:[IPv4:2.24.1.254]
      57.0.0.57          0          100          0          BE
      AS_PATH: 65002 65006
      L2_vni: 24 L3_vni: 0
      ESI : 00.00000000000000000000
6   ARP:[0][0000.abba.baba]:[IPv4:2.24.1.254]
      57.0.0.57          0          100          0          E
      AS_PATH: 65002 65006
      L2_vni: 24 L3_vni: 0
      ESI : 00.00000000000000000000
Route Distinguisher: 6.0.0.6:32793
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes type auto-discovery

Displays information for BGP EVPN auto-discovery routes in the BGP EVPN route table.

## Syntax

**show bgp evpn routes type auto-discovery**

**show bgp evpn routes type auto-discovery brief**

**show bgp evpn routes type auto-discovery detail**

**show bgp evpn routes type auto-discovery esi-value ethernet-tag *tag-id***

## Parameters

### brief

Displays summary information.

### detail

Displays detailed information.

### esi-value

Specifies a 10 byte Ethernet Segment Identifier (ESI) value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH.HH).

### ethernet-tag *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes type auto-discovery** command.

```
device# show bgp evpn routes type auto-discovery

Total number of BGP EVPN AD Routes : 4
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight  Status
Route Distinguisher: 7.0.0.7:1
1       AD:[01.006501e05200000a00] [4294967295]
        23.0.0.23          0          100         0          BME
        AS_PATH: 65002 65078
2       AD:[01.006501e05200000a00] [4294967295]
        23.0.0.23          0          100         0          ME
        AS_PATH: 65002 65078
Route Distinguisher: 8.0.0.8:1
3       AD:[01.006501e05200000a00] [4294967295]
        23.0.0.23          0          100         0          BME
        AS_PATH: 65002 65078
4       AD:[01.006501e05200000a00] [4294967295]
        23.0.0.23          0          100         0          ME
        AS_PATH: 65002 65078
```

The following example shows detailed output for the **show bgp evpn routes type auto-discovery** command.

```
device# show bgp evpn routes type auto-discovery detail

Total number of BGP EVPN AD Routes : 4
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 7.0.0.7:1
1 Prefix: AD:[01.006501e05200000a00][4294967295], Status: BME, Age: 1d5h59m16s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
  Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
  ESI Label Ext Community: 0 All-Active
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
2 Prefix: AD:[01.006501e05200000a00][4294967295], Status: ME, Age: 1d5h59m16s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 3.0.0.3 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
  Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
  ESI Label Ext Community: 0 All-Active
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
Route Distinguisher: 8.0.0.8:1
3 Prefix: AD:[01.006501e05200000a00][4294967295], Status: BME, Age: 1d5h59m16s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
  Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
  ESI Label Ext Community: 0 All-Active
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
4 Prefix: AD:[01.006501e05200000a00][4294967295], Status: ME, Age: 1d5h59m16s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 3.0.0.3 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
  Extended Community: ExtCom:06:01:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
  ESI Label Ext Community: 0 All-Active
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
```

The following example shows summarized output for the **show bgp evpn routes type auto-discovery** command.

```
device# show bgp evpn routes type auto-discovery brief

Total number of BGP EVPN AD Routes : 4
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop          MED          LocPrf          Weight Path
Route Distinguisher: 7.0.0.7:1
*> AD:[01.006501e05200000a00][4294967295]
    23.0.0.23          100          0          65002 65078 ?
* AD:[01.006501e05200000a00][4294967295]
    23.0.0.23          100          0          65002 65078 ?
Route Distinguisher: 8.0.0.8:1
*> AD:[01.006501e05200000a00][4294967295]
    23.0.0.23          100          0          65002 65078 ?
* AD:[01.006501e05200000a00][4294967295]
    23.0.0.23          100          0          65002 65078 ?
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes type ethernet-segment

Displays information for BGP EVPN Ethernet Segment (ES) routes in the BGP EVPN route table.

## Syntax

```
show bgp evpn routes type ethernet-segment
```

```
show bgp evpn routes type ethernet-segment brief
```

```
show bgp evpn routes type ethernet-segment detail
```

```
show bgp evpn routes type ethernet-segment esi-value value { ipv4-address address | ipv6-address address }
```

## Parameters

### brief

Displays summary information.

### detail

Displays detailed information.

### esi-value *value*

Specifies a 10 byte Ethernet Segment Identifier (ESI) value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH.HH).

### ipv4-address *address*

Specifies an IPv4 address.

### ipv6-address *address*

Specifies an IPv6 address.

## Modes

Privileged EXEC mode

## Examples

The following example shows sample output for the **show bgp evpn routes type ethernet-segment** command.

```
device# show bgp evpn routes type ethernet-segment
Total number of BGP EVPN Ethernet Segment Routes : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
Route Distinguisher: 32.0.0.32:1
1      ESR:[00.000000.000000.00aabb] [IPv4:32.0.0.32]
      32.0.0.32      0      100      0      BL
      AS_PATH:
```



## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes type inclusive-multicast

Displays information for BGP EVPN inclusive multicast routes in the BGP EVPN route table.

## Syntax

`show bgp evpn routes type inclusive-multicast`

`show bgp evpn routes type inclusive-multicast brief`

`show bgp evpn routes type inclusive-multicast detail`

`show bgp evpn routes type inclusive-multicast ethernet-tag tag-id ipv4-address address [ l2-vni number ]`

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**ipv4-address** *address*

Specifies an IPv4 address.

**l2-vni** *number*

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

## Modes

Privileged EXEC mode

## Examples

The following example shows sample output for the **show bgp evpn routes type inclusive-multicast** command.

```
device# show bgp evpn routes type inclusive-multicast

Total number of BGP EVPN IMR Routes : 1408
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop      MED           LocPrf        Weight Status
Route Distinguisher: 6.0.0.6:32790
1      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          BE
       AS_PATH: 65002 65006
       L2_vni: 22
2      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          E
       AS_PATH: 65002 65006
       L2_vni: 22
Route Distinguisher: 6.0.0.6:32791
3      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          BE
       AS_PATH: 65002 65006
       L2_vni: 23
4      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          E
       AS_PATH: 65002 65006
       L2_vni: 23
Route Distinguisher: 6.0.0.6:32792
5      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          BE
       AS_PATH: 65002 65006
       L2_vni: 24
6      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          E
       AS_PATH: 65002 65006
       L2_vni: 24
Route Distinguisher: 6.0.0.6:32793
7      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          BE
       AS_PATH: 65002 65006
       L2_vni: 25
8      IMR:[0][IPv4:57.0.0.57]
       57.0.0.57          0             100           0          E
       AS_PATH: 65002 65006
       L2_vni: 25
...
```

show bgp evpn routes type inclusive-multicast

The following example shows summarized output for the **show bgp evpn routes type inclusive-multicast** command.

```
device# show bgp evpn routes type inclusive-multicast brief

Total number of BGP EVPN IMR Routes : 1408
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop      MED      LocPrf    Weight Path
Route Distinguisher: 6.0.0.6:32790
*>  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
*   IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
Route Distinguisher: 6.0.0.6:32791
*>  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
*   IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
Route Distinguisher: 6.0.0.6:32792
*>  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
*   IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
Route Distinguisher: 6.0.0.6:32793
*>  IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
*   IMR:[0][IPv4:57.0.0.57]
      57.0.0.57          100      0      65002 65006 ?
...
```

The following example shows detailed output for the **show bgp evpn routes type inclusive-multicast** command.

```

device# show bgp evpn routes type inclusive-multicast detail

Total number of BGP EVPN IMR Routes : 1408
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:32790
1     Prefix: IMR:[0][IPv4:57.0.0.57], Status: BE, Age: 1d9h45m52s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:22
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000016 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22
2     Prefix: IMR:[0][IPv4:57.0.0.57], Status: E, Age: 1d9h45m52s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:22
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000016 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 22
Route Distinguisher: 6.0.0.6:32791
3     Prefix: IMR:[0][IPv4:57.0.0.57], Status: BE, Age: 1d9h45m52s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:23
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000017 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 23
4     Prefix: IMR:[0][IPv4:57.0.0.57], Status: E, Age: 1d9h45m52s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:23
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000017 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 23
Route Distinguisher: 6.0.0.6:32792
5     Prefix: IMR:[0][IPv4:57.0.0.57], Status: BE, Age: 1d9h45m52s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:24
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000018 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 24
6     Prefix: IMR:[0][IPv4:57.0.0.57], Status: E, Age: 1d9h45m52s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0c:00:00:00:00:00:08 RT 65006:24
      PMSI Attribute Flags: 0x00000000 Label-Stack: 0x00000018 Tunnel-Type: 0x00000006 Tunnel-
IP: 57.0.0.57
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 24
Route Distinguisher: 6.0.0.6:32793
...

```

show bgp evpn routes type inclusive-multicast

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes type ipv4-prefix

Displays information for BGP EVPN IPv4 prefix routes in the BGP EVPN route table.

## Syntax

```
show bgp evpn routes type ipv4-prefix
```

```
show bgp evpn routes type ipv4-prefix brief
```

```
show bgp evpn routes type ipv4-prefix detail
```

```
show bgp evpn routes type ipv4-prefix ip address/mask tag tag-id [ l3vni value ]
```

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

*IPv4 address/mask*

Specifies an IPv4 address and mask.

**tag tag-id**

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**l3vni value**

Specifies a Layer 3 virtual network identifier (VNIs). Valid values range from 1 through 6777215.

## Modes

Privileged EXEC mode

## Examples

The following example shows sample output for the **show bgp evpn routes type ipv4-prefix** command.

```

device# show bgp evpn routes type ipv4-prefix

Total number of BGP EVPN Ipv4Prefix Routes : 1148
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop           MED           LocPrf         Weight Status
Route Distinguisher: 7.0.0.7:2
1      IP4Prefix:[0][2.22.1.0/24]
      23.0.0.23           0             100            0           BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
2      IP4Prefix:[0][2.22.1.0/24]
      23.0.0.23           0             100            0           E
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
3      IP4Prefix:[0][2.23.1.0/24]
      23.0.0.23           0             100            0           BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
4      IP4Prefix:[0][2.23.1.0/24]
      23.0.0.23           0             100            0           E
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
5      IP4Prefix:[0][2.24.1.0/24]
      23.0.0.23           0             100            0           BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
6      IP4Prefix:[0][2.24.1.0/24]
      23.0.0.23           0             100            0           E
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
7      IP4Prefix:[0][2.25.1.0/24]
      23.0.0.23           0             100            0           BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67

```



The following example shows detailed output for the **show bgp evpn routes type ipv4-prefix** command.

```

device# show bgp evpn routes type ipv4-prefix detail

Total number of BGP EVPN Ipv4Prefix Routes : 1148
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 7.0.0.7:2
1      Prefix: IP4Prefix:[0][2.22.1.0/24], Status: BE, Age: 1d10h1m3s
      NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65078
      Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
2      Prefix: IP4Prefix:[0][2.23.1.0/24], Status: E, Age: 1d10h1m2s
      NEXT_HOP: 23.0.0.23, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65078
      Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
3      Prefix: IP4Prefix:[0][2.23.1.0/24], Status: BE, Age: 1d10h1m3s
      NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65078
      Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
4      Prefix: IP4Prefix:[0][2.23.1.0/24], Status: E, Age: 1d10h1m2s
      NEXT_HOP: 23.0.0.23, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65078
      Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
5      Prefix: IP4Prefix:[0][2.24.1.0/24], Status: BE, Age: 1d10h1m3s
      NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65078
      Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
...

```

show bgp evpn routes type ipv4-prefix

The following example shows summarized output for the **show bgp evpn routes type ipv4-prefix** command.

```
device# show bgp evpn routes type ipv4-prefix brief

Total number of BGP EVPN Ipv4Prefix Routes : 1148
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop          MED          LocPrf      Weight Path
Route Distinguisher: 7.0.0.7:2
*> IP4Prefix:[0][2.22.1.0/24]
      23.0.0.23          100           0           65002 65078 ?
*   IP4Prefix:[0][2.22.1.0/24]
      23.0.0.23          100           0           65002 65078 ?
*> IP4Prefix:[0][2.23.1.0/24]
      23.0.0.23          100           0           65002 65078 ?
*   IP4Prefix:[0][2.23.1.0/24]
      23.0.0.23          100           0           65002 65078 ?
*> IP4Prefix:[0][2.24.1.0/24]
      23.0.0.23          100           0           65002 65078 ?
*   IP4Prefix:[0][2.24.1.0/24]
      23.0.0.23          100           0           65002 65078 ?
*> IP4Prefix:[0][2.25.1.0/24]
      23.0.0.23          100           0           65002 65078 ?
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes type ipv6-prefix

Displays information for BGP EVPN IPv6 prefix routes in the BGP EVPN route table.

## Syntax

```
show bgp evpn routes type ipv6-prefix
```

```
show bgp evpn routes type ipv6-prefix brief
```

```
show bgp evpn routes type ipv6-prefix detail
```

```
show bgp evpn routes type ipv6-prefix ipv6 address/mask tag tag-id [ l3vni value ]
```

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

*IPv6address/mask*

Specifies an IPv6 address and mask.

**tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**l3vni** *value*

Specifies a Layer 3 virtual network identifier (VNIs). Valid values range from 1 through 6777215.

## Modes

Privileged EXEC mode

## Examples

The following example shows sample output for the **show bgp evpn routes type ipv6-prefix** command.

```
device# show bgp evpn routes type ipv6-prefix

Total number of BGP EVPN Ipv6Prefix Routes : 1148
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop           MED           LocPrf         Weight Status
Route Distinguisher: 7.0.0.7:2
1      IP6Prefix:[0][2:22:1::/64]
      23.0.0.23           0             100            0            BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
2      IP6Prefix:[0][2:22:1::/64]
      23.0.0.23           0             100            0            E
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
3      IP6Prefix:[0][2:23:1::/64]
      23.0.0.23           0             100            0            BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
4      IP6Prefix:[0][2:23:1::/64]
      23.0.0.23           0             100            0            E
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
5      IP6Prefix:[0][2:24:1::/64]
      23.0.0.23           0             100            0            BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
6      IP6Prefix:[0][2:24:1::/64]
      23.0.0.23           0             100            0            E
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
7      IP6Prefix:[0][2:25:1::/64]
      23.0.0.23           0             100            0            BE
      AS_PATH: 65002 65078
      L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
...
```

The following example shows summarized output for the **show bgp evpn routes type ipv6-prefix** command.

```
device# show bgp evpn routes type ipv6-prefix brief

Total number of BGP EVPN Ipv6Prefix Routes : 1148
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop           MED           LocPrf         Weight Path
Route Distinguisher: 7.0.0.7:2
*> IP6Prefix:[0][2:22:1::/64]
      23.0.0.23           0             100            0            65002 65078 ?
*   IP6Prefix:[0][2:22:1::/64]
      23.0.0.23           0             100            0            65002 65078 ?
*> IP6Prefix:[0][2:23:1::/64]
      23.0.0.23           0             100            0            65002 65078 ?
*   IP6Prefix:[0][2:23:1::/64]
      23.0.0.23           0             100            0            65002 65078 ?
*> IP6Prefix:[0][2:24:1::/64]
      23.0.0.23           0             100            0            65002 65078 ?
*   IP6Prefix:[0][2:24:1::/64]
      23.0.0.23           0             100            0            65002 65078 ?
*> IP6Prefix:[0][2:25:1::/64]
      23.0.0.23           0             100            0            65002 65078 ?
...
```

The following example shows detailed output for the **show bgp evpn routes type ipv6-prefix** command.

```
device# show bgp evpn routes type ipv6-prefix detail

Total number of BGP EVPN Ipv6Prefix Routes : 1148
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 7.0.0.7:2
1 Prefix: IP6Prefix:[0][2:22:1::/64], Status: BE, Age: 1d10h4m16s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
    Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
2 Prefix: IP6Prefix:[0][2:23:1::/64], Status: E, Age: 1d10h4m15s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 3.0.0.3 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
    Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
3 Prefix: IP6Prefix:[0][2:23:1::/64], Status: BE, Age: 1d10h4m16s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
    Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
4 Prefix: IP6Prefix:[0][2:23:1::/64], Status: E, Age: 1d10h4m15s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 3.0.0.3 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
    Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
5 Prefix: IP6Prefix:[0][2:24:1::/64], Status: BE, Age: 1d10h4m16s
  NEXT_HOP: 23.0.0.23, Learned from Peer: 2.0.0.2 (65002)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 65002 65078
    Extended Community: RT 2:2 ExtCom:06:03:50:eb:1a:14:07:67 ExtCom:03:0c:00:00:00:00:08 RT
65078:20
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 2, Admin distance 20
  L3_vni: 20 Router Mac : 50:eb:1a:14:07:67
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes type mac

Displays information for BGP EVPN Media Access Control (MAC) routes in the BGP EVPN route table.

## Syntax

`show bgp evpn routes type mac`

`show bgp evpn routes type mac brief`

`show bgp evpn routes type mac detail`

`show bgp evpn routes type mac mac address ethernet-tag tag-id [ l2-vni number ]`

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

**mac** *mac address*

Specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

**l2-vni**

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

## Modes

Privileged EXEC mode

## Examples

The following example shows sample output for the **show bgp evpn routes type mac** command.

```
device# show bgp evpn routes type mac

show bgp evpn routes type mac
Total number of BGP EVPN MAC Routes : 4216
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
      Prefix          Next Hop          MED          LocPrf        Weight Status
Route Distinguisher: 6.0.0.6:32790
1     MAC:[0][0000.abba.abba]
      57.0.0.57          0            100           0            BE
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
2     MAC:[0][0000.abba.abba]
      57.0.0.57          0            100           0            E
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
3     MAC:[0][0000.abba.baba]
      57.0.0.57          0            100           0            BE
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
4     MAC:[0][0000.abba.baba]
      57.0.0.57          0            100           0            E
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
5     MAC:[0][0027.f8ca.76ba]
      57.0.0.57          0            100           0            BE
      AS_PATH: 65002 65006
      L2_vni: 22
      ESI : 00.00000000000000000000
...
```

The following example shows summarized output for the **show bgp evpn routes type mac** command.

```
device# show bgp evpn routes type mac brief

Total number of BGP EVPN MAC Routes : 4216
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network          Next Hop          MED          LocPrf        Weight Path
Route Distinguisher: 6.0.0.6:32790
*>  MAC:[0][0000.abba.abba]
      57.0.0.57          100           0            65002 65006 ?
*   MAC:[0][0000.abba.abba]
      57.0.0.57          100           0            65002 65006 ?
*>  MAC:[0][0000.abba.baba]
      57.0.0.57          100           0            65002 65006 ?
*   MAC:[0][0000.abba.baba]
      57.0.0.57          100           0            65002 65006 ?
*>  MAC:[0][0027.f8ca.76ba]
      57.0.0.57          100           0            65002 65006 ?
*   MAC:[0][0027.f8ca.76ba]
      57.0.0.57          100           0            65002 65006 ?
Route Distinguisher: 6.0.0.6:32791
*>  MAC:[0][0000.abba.abba]
      57.0.0.57          100           0            65002 65006 ?
*   MAC:[0][0000.abba.abba]
      57.0.0.57          100           0            65002 65006 ?
*>  MAC:[0][0000.abba.baba]
      57.0.0.57          100           0            65002 65006 ?
...
```

The following example shows detailed output for the **show bgp evpn routes type mac** command.

```

device# show bgp evpn routes type mac detail

Total number of BGP EVPN MAC Routes : 4216
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:32790
1     Prefix: MAC:[0][0000.abba.abba], Status: BE, Age: 1d9h58m39s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22
      ESI : 00.000000000000000000
2     Prefix: MAC:[0][0000.abba.abba], Status: E, Age: 1d9h58m40s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 22
      ESI : 00.000000000000000000
3     Prefix: MAC:[0][0000.abba.baba], Status: BE, Age: 1d9h58m39s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22
      ESI : 00.000000000000000000
4     Prefix: MAC:[0][0000.abba.baba], Status: E, Age: 1d9h58m40s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0d:00:00:00:00:00:00 ExtCom:
03:0c:00:00:00:00:00:08 RT 65006:22
      Mac Mobility Sticky: True
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 22
      ESI : 00.000000000000000000
...

```

## History

Release version	Command history
7.0.0	This command was introduced.



# show bgp evpn routes type nd

Displays information for BGP EVPN neighbor-discovery (ND) routes in the BGP EVPN route table.

## Syntax

```
show bgp evpn routes type nd
```

```
show bgp evpn routes type nd brief
```

```
show bgp evpn routes type nd detail
```

```
show bgp evpn routes type nd IPv6 address mac mac address ethernet-tag tag-id
```

## Parameters

**brief**

Displays summary information.

**detail**

Displays detailed information.

*IPv6 address*

Specifies an IPv6 address.

**mac** *mac address*

Specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

**ethernet-tag** *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

## Modes

Privileged EXEC mode

## Examples

The following example shows sample output for the **show bgp evpn routes type nd** command.

```
device# show bgp evpn routes type nd

Total number of BGP EVPN ND Routes : 2808
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix           Next Hop           MED           LocPrf         Weight Status
Route Distinguisher: 6.0.0.6:32790
1 ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
   57.0.0.57           0             100            0             BE
   AS_PATH: 65002 65006
   L2_vni: 22 L3_vni: 0
   ESI : 00.00000000000000000000
2 ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
   57.0.0.57           0             100            0             E
   AS_PATH: 65002 65006
   L2_vni: 22 L3_vni: 0
   ESI : 00.00000000000000000000
3 ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
   57.0.0.57           0             100            0             BE
   AS_PATH: 65002 65006
   L2_vni: 22 L3_vni: 0
   ESI : 00.00000000000000000000
4 ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
   57.0.0.57           0             100            0             E
   AS_PATH: 65002 65006
   L2_vni: 22 L3_vni: 0
   ESI : 00.00000000000000000000
Route Distinguisher: 6.0.0.6:32791
5 ND:[0][0000.abba.abba]:[IPv6:2:23:1::254]
   57.0.0.57           0             100            0             BE
   AS_PATH: 65002 65006
   L2_vni: 23 L3_vni: 0
   ESI : 00.00000000000000000000
6 ND:[0][0000.abba.abba]:[IPv6:2:23:1::254]
   57.0.0.57           0             100            0             E
   AS_PATH: 65002 65006
   L2_vni: 23 L3_vni: 0
   ESI : 00.00000000000000000000
...
```

The following example shows summarized output for the **show bgp evpn routes type nd** command.

```
device# show bgp evpn routes type nd brief

Total number of BGP EVPN ND Routes : 2808
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network           Next Hop           MED           LocPrf         Weight Path
Route Distinguisher: 6.0.0.6:32790
*> ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
   57.0.0.57           0             100            0             65002 65006 ?
* ND:[0][0000.abba.abba]:[IPv6:2:22:1::254]
   57.0.0.57           0             100            0             65002 65006 ?
*> ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
   57.0.0.57           0             100            0             65002 65006 ?
* ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
   57.0.0.57           0             100            0             65002 65006 ?
Route Distinguisher: 6.0.0.6:32791
*> ND:[0][0000.abba.abba]:[IPv6:2:23:1::254]
   57.0.0.57           0             100            0             65002 65006 ?
* ND:[0][0000.abba.abba]:[IPv6:2:23:1::254]
   57.0.0.57           0             100            0             65002 65006 ?
*> ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba]
   57.0.0.57           0             100            0             65002 65006 ?
...
```

The following example shows detail output for the **show bgp evpn routes type nd** command.

```

device# show bgp evpn routes type nd detail

Total number of BGP EVPN ND Routes : 2808
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
      E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
      S:SUPPRESSED F:FILTERED s:STALE
Route Distinguisher: 6.0.0.6:32790
1      Prefix: ND:[0][0000.abba.abba]:[IPv6:2:22:1::254], Status: BE, Age: 1d10h2m16s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22 L3_vni: 0
      ESI : 00.000000000000000000000000
2      Prefix: ND:[0][0000.abba.abba]:[IPv6:2:22:1::254], Status: E, Age: 1d10h2m17s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
      Default Extd Gw Community: Received
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 22 L3_vni: 0
      ESI : 00.000000000000000000000000
3      Prefix: ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba], Status: BE, Age: 1d10h2m13s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 2.0.0.2 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
      Mac Mobility Sticky: True
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      Adj_RIB_out count: 2, Admin distance 20
      L2_vni: 22 L3_vni: 0
      ESI : 00.000000000000000000000000
4      Prefix: ND:[0][0027.f8ca.76ba]:[IPv6:fe80::227:f8ff:feca:76ba], Status: E, Age: 1d10h2m13s
      NEXT_HOP: 57.0.0.57, Learned from Peer: 3.0.0.3 (65002)
      LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
      AS_PATH: 65002 65006
      Extended Community: ExtCom:06:00:01:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT
65006:22
      Mac Mobility Sticky: True
      Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
      L2_vni: 22 L3_vni: 0
      ESI : 00.000000000000000000000000
Route Distinguisher: 6.0.0.6:32791
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn routes unreachable

Displays route information about BGP EVPN routes whose destinations are unreachable through any of the paths in the BGP EVPN route table.

## Syntax

```
show bgp evpn routes unreachable
```

```
show bgp evpn routes unreachable type { arp | auto-discovery | ethernet-segment | inclusive-multicast | ipv4-prefix | ipv6-prefix | mac | nd }
```

## Parameters

### type

Specifies the type of route.

### arp

Specifies Address Resolution Protocol (ARP) routes.

### auto-discovery

Specifies automatically discovered routes.

### ethernet-segment

Specifies Ethernet Segments (ES) routes.

### inclusive-multicast

Specifies inclusive multicast routes.

### ipv4-prefix

Specifies IPv4 prefix routes.

### ipv6-prefix

Specifies IPv6 prefix routes.

### mac

Specifies MAC routes.

### nd

Specifies neighbor discovery (ND) routes.

## Modes

Privileged EXEC mode

## Examples

The following example shows output for the **show bgp evpn routes unreachable** command.

```
device# show bgp evpn routes unreachable

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight      Status
1  AD:[00.000000000000989900][4294967295]
      78.0.0.78      0      100      0      BME
      AS_PATH: 2 3
2  AD:[00.000000000000989900][4294967295]
      78.0.0.78      0      100      0      ME
      AS_PATH: 2 3
3  ESR:[00.000000.000000.989900][IPv4:8.0.0.8]
      78.0.0.78      0      100      0      BE
      AS_PATH: 2 3
4  ESR:[00.000000.000000.989900][IPv4:8.0.0.8]
      78.0.0.78      0      100      0      E
      AS_PATH: 2 3
5  IP4Prefix:[0][11.1.1.0/24]
      78.0.0.78      0      100      0      BE
      AS_PATH: 2 3
      L3_vni: 10020 Router Mac : 50:eb:1a:13:ce:f5
6  IP4Prefix:[0][11.1.1.0/24]
      78.0.0.78      0      100      0      E
      AS_PATH: 2 3
      L3_vni: 10020 Router Mac : 50:eb:1a:13:ce:f5
7  IP4Prefix:[0][12.1.1.0/24]
      78.0.0.78      0      100      0      BE
      AS_PATH: 2 3
      L3_vni: 10020 Router Mac : 50:eb:1a:13:ce:f5
8  IP4Prefix:[0][12.1.1.0/24]
      78.0.0.78      0      100      0      E
      AS_PATH: 2 3
      L3_vni: 10020 Router Mac : 50:eb:1a:13:ce:f5
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bgp evpn summary

Displays summarized information for BGP EVPN.

## Syntax

```
show bgp evpn summary
```

## Modes

Privileged EXEC mode

## Examples

The following example shows summarized information for BGP EVPN.

```
device# show bgp evpn summary

BGP4 Summary
Router ID: 6.0.0.6   Local AS Number: 65006
Confederation Identifier: not configured
Confederation Peers:
Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
Number of Neighbors Configured: 4, UP: 3
Number of Routes Installed: 22143, Uses 2657160 bytes
Number of Routes Advertising to All Neighbors: 25620 (25620 entries), Uses 1537200 bytes
Number of Attribute Entries Installed: 2892, Uses 332580 bytes
Neighbor Address  AS#           State      Time      Rt:Accepted  Filtered  Sent      ToSend
2.6.0.0          65002         CONN       18h 8m16s  0            0         0         8153
3.6.0.0          65002         ESTAB      18h 7m29s  6971         3         2460      0
4.6.0.0          65002         ESTAB      18h 7m29s  7004         7         7370      0
5.6.0.0          65002         ESTAB      18h 7m29s  7007         8         7637      0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show bpdu-drop

Displays information about BPDU guard.

## Syntax

```
show bpdu-drop [ interface { port-channel num | <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### interface

Selects an interface (required).

### port-channel *num*

Selects a port channel interface. The number of available channels ranges from 1 through 6144

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command can be entered on any RBridge in an Extreme VCS Fabric.

## Related Commands

[capture packet interface](#)

# show capture packet interface

Displays information about captured packets.

## Syntax

```
show capture packet interface { all | <N>gigabitethernet rbridge-id/slot/port | port-channel number | ve vlan_id }
```

## Parameters

**all**

Selects all interfaces.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies the port-channel number. The number of available channels ranges from 1 through 6144.

**ve** *vlan\_id*

Specifies the virtual Ethernet interface. Range is from 1 through 8191.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays information captured by means of the **capture packet interface** command

This command can be entered on any RBridge.



## Examples

To view information about captured packets on all interfaces:

```
switch# show capture packet interface all
Packet Capture configured on the following interfaces
Te 130/0/5 >> ISL
Te 130/0/6 >> ISL
Te 130/0/21
Te 130/0/23
Te 130/0/24
Frame Received Time : Sat Mar 9 2013 0:57:0:282
Packet Type          : ELD
Packet Direction     : TX
Interface info       : Te 130/0/21
ETHERNET HEADER
SrcMAC               : 00:05:33:5e:01:67
DstMAC               : 03:05:33:5d:f3:fa
Ethtype              : 0x8100
Eth Frametype        : 0x33
VlanID               : 0xffff
ELD PAYLOAD DETAILS
-----
Vlan id              : 2
Src-Rbridgeid        : 130
Src-Priority          : 5
Magic Number         : 5103
```

To view information about captured packets on a VE interface:

```
device# show capture packet int ve 20
Frame Received Time : Fri Jul 22 2016 17:42:46:653
Packet Type          : ICMP
Packet Direction     : TX
Interface info       : V1 20
ETHERNET HEADER
SrcMAC               : 00:27:f8:bb:5f:68
DstMAC               : 50:eb:1a:36:33:29
Tag Protocol ID      : 0x8100
VlanID               : 0x14
Ethtype              : 0x800
Internet proto,SrcIP : 20.1.1.1, DstIP: 30.1.1.1
Interface            : V1 20
Type of service      : 0
Length               : 84
Identification       : 15317
Fragmentation        : 00 00
TTL                  : 64
protocol              : 1
Checksum             : 0a d1
ICMP Type            : 0 Echo Reply
ICMP Code             : 0
```

show capture packet interface

To view information about captured packets on a port-channel interface:

```
device# show capture packet interface port-channel 13
Frames Logged on interface Po 13 :
```

```
-----
Frame Received Time : Fri Jul 22 2016 18:10:51:202
Packet Type         : ICMP
Packet Direction    : RX
Interface info      : Po 13
ETHERNET HEADER
SrcMAC               : 50:eb:1a:36:33:6d
DstMAC               : 00:27:f8:bb:5f:ac
Ethtype              : 0x800
Internet proto,SrcIP : 13.1.1.2, DstIP: 20.1.1.1
Interface            : Po 13
Type of service      : 0
Length              : 84
Identification       : 0
Fragmentation        : 40 00
TTL                  : 64
protocol             : 1
Checksum             : 17 a5
ICMP Type            : 8 Echo Request
ICMP Code            : 0
```

## History

Release version	Command history
7.1.0	This command was modified to support port-channel and VE interfaces.

## Related Commands

[capture packet interface](#)

# show cee maps

Displays information on the defined CEE maps. The configuration state is displayed with a list of all of the Layer 2 interfaces bound to the CEE map.

## Syntax

```
show cee maps default
```

## Command Default

The only map name allowed is "default."

## Modes

Privileged EXEC mode

## Usage Guidelines

Network OS only allows the CEE map named "default."

## Examples

To view the CEE map:

```
switch0# show cee maps

CEE Map 'default'
Precedence: 1
Remap Fabric-Priority to Priority 0
Remap Lossless-Priority to Priority 0
Priority Group Table
1:  Weight 40, PFC Enabled, BW% 40
2:  Weight 40, PFC Disabled, BW% 40
3:  Weight 20, PFC Disabled, BW% 20
15.0: PFC Disabled
15.1: PFC Disabled
15.2: PFC Disabled
15.3: PFC Disabled
15.4: PFC Disabled
15.5: PFC Disabled
15.6: PFC Disabled
15.7: PFC Disabled
Priority Table
CoS:   0   1   2   3   4   5   6   7
-----
PGID:  2   2   3   1   2   2   2 15.0
```

## Related Commands

[cee, cee-map \(configuration\)](#)

# show cert-util ldapca

Displays the Lightweight Directory Access Protocol (LDAP) Certification Authority (CA) certificate.

## Syntax

```
show cert-util ldapca [ rbridge-id { rbridge_id | all } ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

To display the LDAP certificate on the switch:

```
switch: show cert-util syslogcacert rbridge-id 3
```

# show cert-util sshkey

Displays the public SSH key for a specified user..

## Syntax

```
show cert-util sshkey user user_id [ rbridge-id { rbridge-id | all } ]
```

## Parameters

**user** *user\_id*

The user ID to display.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

A typical output of this command:

```
switch# show cert-util sshkey user testuser

user's public keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAtTCFzC1lfjwV9hjdTqv2ulSvmsmf7q7MS92Ctc3pDje/
YGYJPHVUi8bQX0XAsCAuzdsZL0BlVHdYP0lL4HStuIo8okfn4xLxrazqzwVeeL8p5Zcspf9zK8HmDzNpZ/
OuQ9MvfOuzbseYrovqgYLFgfPvY6vleFXZo6lvVncFM7uFzasED9o9JUSBRORhBki7vB0SG69yNn6ADnmpQW6QOu
+nYuZaWX00QXk2OIB+hidjxSQVAFVLidSIGyfDD0go
+JAE3osxZxwQa5jcorASs4q2Gt4tSYERpvzOsjaAR5YivbmmBTIQWdUuR9Laz8s8VKF4Di9HQ4kE+xyBeAFNvQ==
bmeenaks@blc-10-6
```

To see the output of rbridge-id 3:

```
switch# show cert-util sshkey user testuser rbridge-id 3
```

## Related Commands

[certutil import sshkey](#)

show cert-util syslogca

# show cert-util syslogca

Displays the syslog Certification Authority (CA) certificate.

## Syntax

```
show cert-util syslogca [ rbridge-id { rbridge_id | all } ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

To display the LDAP certificate on the switch:

```
switch# show cert-util ldapca
```

```
LDAP CA
```

# show chassis

Displays the Field Replaceable Unit (FRU) header content for each object in the chassis.

## Syntax

```
show chassis [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID for the switch.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is executed on the local switch and is supported only on the local switch. The output of this command depends on the platforms on which it is executed. Not all information is available for all platforms. In cases where information is not available, the lines are suppressed.

Pagination is not supported with this command. Use the "more" parameter to display the output one page at a time.

## Command Output

The **show chassis** command displays the Field Replaceable Unit (FRU) header content for each object in the chassis.

Output field	Description
Chassis name and model	For example, BR-VDX6740-48
Chassis backplane revision	
Object type	MM (management module), SFM (switch fabric module), LC (line card), CHASSIS, FAN, POWER SUPPLY, SW CID (chassis ID), WWN (world wide name), or UNKNOWN
Object number	Slot number (for blades), Unit number (for everything else)
Brief description	If the FRU is part of an assembly
FRU header version number or blade version	Header Version: x
Maximum allowed power consumption	Positive value for power supplies; negative value for consumers
Real-time power usage (W)	For FRUs that support real-time power management
Part number (up to 14 characters)	Factory Part Num: xx-xxxxxx-x x
Serial number (up to 12 characters)	Factory Serial Num: xxxxxxxxx
FRU manufacture date	Manufacture: Day: dd Month: mm Year: yyyy
Date of the last FRU header update	Update: Day: dd Month: mm Year: yyyy
Cumulative time (days) FRU inserted in the chassis with Network OS running	Time Alive :dd days

Output field	Description
Current time (days) since FRU was last powered on or the system restarted	Time Awake: dd days
Airflow direction	
Externally supplied ID (up to 10 characters)	ID: xxxxxxxxxx
Externally supplied part number (up to 20 characters)	Part Num: xxxxxxxxxxxxxxxxxxxx
Externally supplied serial number (up to 20 characters)	Serial Num: xxxxxxxxxxxxxxxxxxxx
Externally supplied revision number (up to 4 characters)	Revision Num: xxxx

## Examples

To display the FRU information on an Extreme VDX device:

```
device# show chassis rbridge-id 54
```

```
Chassis Name:          BR-VDX6740
Chassis Backplane Revision: 2
switchType: 96
FAN Unit: 1
Time Awake:           64 days
FAN Unit: 2
Time Awake:           64 days
POWER SUPPLY Unit: 1
Header Version:       2
Factory Part Num:     40-1000590-03
Factory Serial Num:   BWU0406G006
Manufacture:          Day: 18 Month: 2 Year: 2011
Update:               Day: 1 Month: 7 Year: 2012
Time Alive:           594 days
Time Awake:           0 days
POWER SUPPLY Unit: 2
Header Version:       2
Factory Part Num:     40-1000590-03
Factory Serial Num:   BWU0406G006
Manufacture:          Day: 18 Month: 2 Year: 2011
Update:               Day: 1 Month: 7 Year: 2012
Time Alive:           594 days
Time Awake:           64 days
CHASSIS/WWN Unit: 1
Header Version:       2
Power Consume Factor: 0
Factory Part Num:     40-1000590-03
Factory Serial Num:   BWU0406G006
Manufacture:          Day: 18 Month: 2 Year: 2011
Update:               Day: 1 Month: 7 Year: 2012
Time Alive:           594 days
Time Awake:           64 days
```



To display the FRU information on an Extreme VDX 8770-4:

```

device# show chassis rbridge-id 1

Chassis Name:          BR-VDX8770-4
Chassis Backplane Revision: 2
switchType: 1000
MM Slot: M1
Blade Version:        3
Power Consume Factor: -120
Power Usage (Watts):  -43
Factory Part Num:     60-1002179-07
Factory Serial Num:   BVT0329G00D
Manufacture:          Day: 26 Month: 7 Year: 11
Update:               Day: 30 Month: 6 Year: 2012
Time Alive:           78 days
Time Awake:           1 days
SFM Slot: S2
Blade Version:        3
Power Consume Factor: -150
Power Usage (Watts):  -132
Factory Part Num:     60-1002180-05
Factory Serial Num:   BVU0321G01F
Manufacture:          Day: 39 Month: 5 Year: 17
Update:               Day: 30 Month: 6 Year: 2012
Time Alive:           76 days
Time Awake:           1 days
LC Slot: L1
Blade Version:        3
Power Consume Factor: -400
Factory Part Num:     60-1002181-08
Factory Serial Num:   BVV0333G00E
Manufacture:          Day: 17 Month: 8 Year: 11
Update:               Day: 30 Month: 6 Year: 2012
Time Alive:           69 days
Time Awake:           1 days
LC Slot: L2
Blade Version:        3
Power Consume Factor: -400
Factory Part Num:     60-1002181-07
Factory Serial Num:   BVV0326G01B
Manufacture:          Day: 5 Month: 7 Year: 11
Update:               Day: 30 Month: 6 Year: 2012
Time Alive:           75 days
Time Awake:           1 days
LC Slot: L3
Blade Version:        3
Power Consume Factor: -400
Power Usage (Watts):  -261
Factory Part Num:     40-1000573-01
Factory Serial Num:   BTF0333G002
Manufacture:          Day: 48 Month: 8 Year: 17
Update:               Day: 30 Month: 6 Year: 2012
Time Alive:           58 days
Time Awake:           1 days
LC Slot: L4
Blade Version:        3
Power Consume Factor: -400
Factory Part Num:     60-1002181-07
Factory Serial Num:   BVV0326G01A
Manufacture:          Day: 5 Month: 7 Year: 11
Update:               Day: 30 Month: 6 Year: 2012
Time Alive:           80 days
Time Awake:           1 days
POWER SUPPLY Unit: 1
Power Consume Factor: 3000
Factory Part Num:     23-0000135-01
Factory Serial Num:   BMM2J02G003
Manufacture:          Day: 1 Month: 1 Year: 2011
Time Awake:           1 days
ID:                   LPCS

```

show chassis

```
Part Num:          SP750Z1A
Rework:           A
POWER SUPPLY Unit: 2
Power Consume Factor: 3000
Factory Part Num:  23-0000135-01
Factory Serial Num: BMM2J02G008
Manufacture:       Day: 1 Month: 1 Year: 2011
Time Awake:       1 days
ID:               LPCS
Part Num:         SP750Z1A
Rework:           A
FAN Unit: 1
Power Consume Factor: -126
Power Usage (Watts): -19
Factory Part Num:  60-1002130-02
Factory Serial Num: BYX0320G007
Manufacture:       Day: 3 Month: 6 Year: 17
Time Awake:       1 days
FAN Unit: 2
Power Consume Factor: -126
Power Usage (Watts): -21
Factory Part Num:  60-1002130-02
Factory Serial Num: BYX0320G011
Manufacture:       Day: 3 Month: 6 Year: 17
Time Awake:       1 days
CID Unit: 1
Power Consume Factor: -1
Factory Part Num:  60-1002178-01
Factory Serial Num: BWF0319G015
Manufacture:       Day: 3 Month: 6 Year: 17
Time Awake:       1 days
CID Unit: 2
Power Consume Factor: -1
Factory Part Num:  60-1002178-01
Factory Serial Num: BWF0319G01Z
Manufacture:       Day: 3 Month: 6 Year: 17
Time Awake:       1 days
Chassis Factory Serial Num: BZA0320G00W
```

## Related Commands

[show linecard](#), [show mm](#), [show sfm](#)

# show cipherset

Displays the current cipherset status for LDAP and SSH.

## Syntax

```
show cipherset
```

## Modes

Privileged EXEC mode

## Examples

To display cipherset status on the switch:

```
switch# show cipherset
```

```
LDAP Cipher List      : !DH:HIGH:-MD5
SSH Cipher List       : 3des-cbc, aes128-cbc, aes192-cbc, aes256-cbc
```

## Related Commands

[cipherset](#)

# show class-maps

Displays all the class-maps configured in the system.

## Syntax

```
show class-maps
```

## Modes

Privileged EXEC mode

## Related Commands

[class-map](#), [policy-map](#)

# show cli

Displays all the current CLI settings.

## Syntax

`show cli`

## Modes

Privileged EXEC mode

## Examples

Typical command output display.

```
switch# show cli
autowizard                false
complete-on-space        false
history                   100
idle-timeout              600
ignore-leading-space      false
output-file               terminal
paginate                  true
prompt1                   \H\M#
prompt2                   \H(\m) #
screen-length             73
screen-width              120
service prompt config     true
show-defaults             false
terminal                  ansi
```

# show cli history

Displays the last 512 commands executed on the local node across user sessions.

## Syntax

`show cli history`

## Modes

Privileged EXEC mode

# show clock

Returns the local time, date, and time zone.

## Syntax

```
show clock [ rbridge-id { rbridge-id | all } ]
```

## Command Default

The local clock is used.

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the RBridge ID is not provided, status results default to the local switch (LOCL). If **rbridge-id all** is executed, the command displays the status for all switches in the cluster.

This command is currently supported only on the local RBridge.

## Examples

The following example shows the clock time for all switches in the cluster.

```
device# show clock rbridge-id all
```

The following example shows clock time for switch with rbridge-id 16.

```
device# show clock rbridge-id 16
```

## Related Commands

[clock set](#), [clock timezone \(Privileged EXEC mode\)](#), [clock timezone \(RBridge ID configuration mode\)](#), [ntp server](#)

# show config snapshot

Displays the snapshots present on the switch.

## Syntax

```
show config snapshot [ rbridge-id { rbridge-id | all } ] [ snapshot-id ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

*snapshot-id*

Specifies the name of the snapshot that has been captured. This can be any combination of characters and numbers. The range is from 1 through 50.

## Modes

Privileged EXEC mode

## Usage Guidelines

A maximum of four snapshots for each RBridge ID can be stored on the switch.

## Related Commands

[copy snapshot](#)



# show copy-support status

Displays the status of the copy support operation.

## Syntax

```
show copy-support status [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

The status is indicated by the percentage of completion. On a modular chassis, Use this command to display status information for each module along with the slot number and SS type. NORMAL indicates process is proceeding or completed without errors. FAULTY indicates a faulty blade.

This command is supported only on the local switch.

## Examples

To display the support upload status on an Extreme VDX 8770-4:

```
device# show copy-support status
```

```
Slot Name          SS type          Completion Percentage
#####
M1                 NORMAL           [100%]
M2                 NORMAL           [100%]
L1/0               NORMAL           [100%]
L1/1               NORMAL           [100%]
L2/0               NORMAL           [100%]
L2/1               NORMAL           [100%]
L4/0               NORMAL           [100%]
L4/1               NORMAL           [100%]
```

## Related Commands

[copy support](#), [copy support-interactive](#), [show support](#)

# show crypto ca

Displays the crypto trust point/certificate information for HTTPS.

## Syntax

```
show crypto ca {trustpoint | certificates} rbridge-id {rbridge-id | all}
```

## Parameters

### trustpoint

Displays the trustpoint and associated key pair details.

### certificates

Displays the CA certificate and Identity certificate details.

### rbridge-id {rbridge-id | all}

If unspecified, executes only for current node. If a particular rbridge-id is specified then the command is executed for that node. If **rbridge-id** all is specified, the command executes for all nodes in the cluster.

## Modes

Privileged EXEC mode

## Usage Guidelines

To execute this command from other configuration modes, use the **do** command modifier.

## Examples

Typical command display output:

```
device# show crypto ca trustpoint
rbridge-id:1
trustpoint: t1; key-pair: k1
```

Typical command display output for certificates:

```
device# show crypto ca certificates
rbridge-id:1
Trustpoint: t1
certificate:
SHA1 Fingerprint=B7:5B:DB:9B:24:69:40:39:36:66:4D:59:2C:69:83:8E:93:CA:23:0C
Subject: C=US, ST=CA, L=SJ, O=BRC, OU=SF, CN=10:00:00:27:F8:87:70:29
Issuer: C=US, ST=CA, L=SJ, O=BR, OU=SF, CN=SOUND/emailAddress=sravi
Not Before: Oct 6 23:44:27 2014 GMT
Not After : Oct 6 23:44:27 2015 GMT
purposes: sslserver
CA certificate:
SHA1 Fingerprint=76:5B:D4:2C:CB:54:FE:6B:C5:E0:E3:FD:11:B0:88:70:80:12:C6:63
Subject: C=US, ST=CA, L=SJ, O=BR, OU=SF, CN=SOUND/emailAddress=sravi
Issuer: C=US, ST=CA, L=SJ, O=BR, OU=SF, CN=SOUND/emailAddress=sravi
Not Before: Sep 19 20:56:49 2014 GMT
Not After : Oct 19 20:56:49 2014 GMT
purposes: sslserver
```

## History

Release version	Command history
6.0.0	This command was introduced.

# show crypto key

Displays the crypto key pair information for HTTPS.

## Syntax

```
show crypto key mypubkey rbridge-id {rbridge-id | all}
```

## Parameters

**rbridge-id** {rbridge-id | all}

If unspecified, executes only for current node. If a particular rbridge-id is specified then the command is executed for that node.

## Modes

Privileged EXEC mode

## Usage Guidelines

To execute this command from other configuration modes, use the **do** command modifier.

## Examples

Typical command output:

```
device# show crypto key mypubkey
rbridge-id:1
key type: ecdsa
key label: k1
key size: 384
```

## History

Release version	Command history
6.0.0	This command was introduced.

# show dadstatus

Displays the current DHCP Automatic Deployment (DAD) status output on the switch.

## Syntax

```
show dadstatus
```

## Modes

Privileged EXEC mode

## Usage Guidelines

When run from the principal node, the DAD progress for all nodes is displayed. When run from the secondary node the progress for the current node is displayed.

If DAD fails, one of the following errors will show in the output:

1. DHCP auto-deployment failed during DHCP process
2. DHCP auto-deployment failed in sanity check
3. DHCP auto-deployment failed due to same firmware
4. DHCP auto-deployment failed to start firmware download
5. DHCP auto-deployment failed due to firmware download failure

## Examples

The following example displays DAD status output on the switch:

```
device# show dadstatus
DAD status:      Enabled/Failed/Complete, DAD-13xx
Principal rbridge id:  2
Total nodes:     4
Nodes in cluster:  3
Rbridge id      Firmware      Status
1               nos6.0.1      In cluster
2               nos6.0.1      Succeed
3               nos6.0.1      Running script
4               Unknown      Unknown
```

## Related Commands

[firmware download](#), [firmware download logical-chassis](#), [dhcp auto-deployment enable](#)

show debug dhcp packet

# show debug dhcp packet

Displays the DHCP packet capture configuration for interfaces configured for DHCP packet capturing.

## Syntax

```
show debug dhcp packet
```

## Modes

Privileged EXEC mode

## Examples

```
sw0# show debug dhcp packet

% DHCP protocol RCV debug is enabled on interface Te 3/18
% DHCP protocol TX debug is enabled on interface Te 3/18
PCAP Buffer Configuration for Vrf ID 0: Buffer Type is Linear and BufferSize is 2056
```

## Related Commands

[debug dhcp packet buffer interface](#)

# show debug dhcp packet buffer

Displays DHCP packets saved in the DHCP packet capture buffer for all VRF IDs.

## Syntax

```
show debug dhcp packet buffer
```

## Modes

Privileged EXEC mode

## Examples

The following command displays buffer content for all VRF IDs.

```
sw0# show debug dhcp packet buffer
Protocol Type      : DHCP
Packet Flow       : RX
Src Port          : 68 (DHCP Client)
Dst Port          : 67 (DHCP Server)
Message Type      : 1 (DHCP-Discover)
Hardware Type     : 1 (Ethernet (10Mb))
Hw Address Len   : 6
Hops              : 0
Transaction ID    : 0
Seconds Elapsed  : 0
BootP Flags      : 8000
Client IP         : 0.0.0.0
Your (client) IP : 0.0.0.0
Next Server IP   : 0.0.0.0
Relay Agent IP   : 0.0.0.0
Client MAC Add   : 00:10:94:00:00:01
Server Host Name : Not Given
Boot File Name   : Not Given
*****
Protocol Type     : DHCP
Packet Flow      : TX
Src Port         : 67 (DHCP Server)
Dst Port        : 68 (DHCP Client)
Message Type     : 2 (DHCP-Offer)
Hardware Type    : 1 (Ethernet (10Mb))
Hw Address Len  : 6
Hops             : 1
Transaction ID   : 0
Seconds Elapsed : 0
BootP Flags     : 8000
Client IP       : 0.0.0.0
Your (client) IP : 10.10.10.30
Next Server IP  : 20.20.20.20
Relay Agent IP  : 10.10.10.10
Client MAC Add  : 00:10:94:00:00:01
Server Host Name : Not Given
Boot File Name  : Not Given
*****
Protocol Type     : DHCP
Packet Flow      : RX
Src Port         : 68 (DHCP Client)
Dst Port        : 67 (DHCP Server)
Message Type     : 3 (DHCP-Request)
Hardware Type    : 1 (Ethernet (10Mb))
Hw Address Len  : 6
Hops             : 0
Transaction ID   : 0
Seconds Elapsed : 0
BootP Flags     : 8000
Client IP       : 0.0.0.0
Your (client) IP : 0.0.0.0
Next Server IP  : 0.0.0.0
Relay Agent IP  : 0.0.0.0
Client MAC Add  : 00:10:94:00:00:01
Server Host Name : Not Given
Boot File Name  : Not Given
*****
Protocol Type     : DHCP
Packet Flow      : TX
Src Port         : 67 (DHCP Server)
Dst Port        : 68 (DHCP Client)
Message Type     : 5 (DHCP-Ack)
Hardware Type    : 1 (Ethernet (10Mb))
Hw Address Len  : 6
Hops             : 1
Transaction ID   : 0
```



```
Seconds Elapsed      : 0
BootP Flags          : 8000
Client IP            : 0.0.0.0
Your (client) IP     : 10.10.10.30
Next Server IP       : 20.20.20.20
Relay Agent IP       : 10.10.10.10
Client MAC Add       : 00:10:94:00:00:01
Server Host Name     : Not Given
Boot File Name       : Not Given
*****
```

## Related Commands

[debug dhcp packet buffer clear](#), [debug dhcp packet buffer interface](#)

```
show debug ip bgp all
```

# show debug ip bgp all

Displays all BGP4 debug options that are enabled.

## Syntax

```
show debug ip bgp all
```

## Modes

Privileged EXEC mode

## Examples

```
switch# show debug ip bgp all
```

# show debug ip igmp

Displays the IGMP packets received and transmitted, as well as related events.

## Syntax

```
show debug ip igmp
```

## Modes

Privileged EXEC mode

## Examples

The following example displays the status of the **show debug ip igmp** command.

```
device# show debug ip igmp
```

```
IGMP debugging status: rbridge_id 2
```

```
-----
```

```
errors          : off
group           : off
packets         : off
query           : off
report          : off
direction       : none
vlan            : none
l2_port         : none
```

## History

Release version	Command history
7.0.0	This command was modified to include the output example.

# show debug ip pim

Displays the current state of the Protocol Independent Multicast (PIM) debug flags.

## Syntax

`show debug ip pim`

## Modes

Privileged EXEC mode

## Examples

A typical output of this command.

```
switch# show debug ip pim

PIM debugging status:
-----
add-del-oif   : off
bootstrap    : off
group        : off
join-prune   : on
nbr-change   : off
packets      : off
parent       : off
regproc      : off
route-change : off
rp           : off
source       : off
-----
```

# show debug ipv6 mld

Displays the IPv6 Multicast Listener Discovery (MLD) packets received and transmitted, as well as related events.

## Syntax

```
show debug ipv6 mld [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the output of the **show debug ipv6 mld** command.

```
device# show debug ipv6 mld
MLD debugging status: rbridge_id 2
-----
errors          : on
group           : on (ff01::1)
packets         : off
query          : off
report          : off
direction       : none
vlan            : none
l2_port         : none
```

The following example displays the output of the **show debug ipv6 mld** for a specific RBridge.

```
device# show debug ipv6 mld rbridge-id 3
MLD debugging status: rbridge_id 3
-----
errors          : off
group           : off
packets         : on
query          : on
report          : on
direction       : none
vlan            : none
l2_port         : none
```

The following example displays the status of the **show debug ipv6 mld** command for a specific RBridge that is either nonexistent, disconnected, not yet in the cluster.

```
device# show debug ipv6 mld rbridge-id 8
% Error: Rbridge does not exist or node not connected or cluster formation is happening.
```

The following example displays the status of the **show debug ipv6 mld** command for all RBridges.

```
device# show debug ipv6 mld rbridge-id all
MLD debugging status: rbridge_id 2
-----
errors          : on
group           : on (ff01::1)
packets         : off
query          : off
report          : off
direction      : none
vlan           : none
l2_port        : none
MLD debugging status: rbridge_id 3
-----
errors          : off
group           : off
packets         : on
query          : on
report          : on
direction      : none
vlan           : none
l2_port        : none
```

## History

Release version	Command history
7.1.0	This command was modified to include the <b>rbridge-id</b> keyword and examples.

# show debug ipv6 packet

Displays IPv6 packets captured through the packet capture utility on an interface or all interfaces, as well as the packet capture configuration on the switch.

## Syntax

```
show debug ipv6 packet [ buffer [ all | interface [<N> gigabitethernet rbridge-id/slot/port | ve vlan_id ] ] [ rx | tx ]
```

## Parameters

### buffer

Specifies IPv6 packets.

### all

Specifies all interfaces.

### interface

Specifies an interface.

### <N> gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

### ve *vlan\_id*

Specifies a virtual Ethernet interface.

## Command Default

None

## Modes

Privileged EXEC mode

## Examples

To display the current PCAP configuration on the switch:

```
switch# show debug ipv6 packet
```

show debug ipv6 packet

To display IPv6 packets captured on all interfaces:

```
switch# show debug ipv6 packet buffer all
```

To display IPv6 packets captured on a specific interface:

```
switch# show debug ipv6 packet buffer int te 54/0/1
```

## History

Release version	Command history
5.0.0	This command was introduced.



# show debug lacp

Displays the status of LACP debugging flags on the switch.

## Syntax

```
show debug lacp
```

## Modes

Privileged EXEC mode

# show debug lldp

Displays the LLDP debugging status on the switch.

## Syntax

```
show debug lldp
```

## Modes

Privileged EXEC mode

## Examples

To display the LLDP debugging status on the switch:

```
switch# show debug lldp
```

```
LLDP debugging status:  
Interface te0/0      : Transmit Receive  Detail
```

# show debug spanning-tree

Displays the status of STP debugging flags on the switch.

## Syntax

```
show debug spanning-tree
```

## Modes

Privileged EXEC mode

# show debug udd

Shows UDLD debug status on the switch.

## Syntax

```
show debug udd
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays the unidirectional link detection (UDLD) protocol debug status of the switch. The status reflects the debugging you set with the **debug udd** command.

## Related Commands

[protocol udd](#), [debug udd packet](#)

# show debug vrrp

Displays the status of VRRP debugging flags on the switch.

## Syntax

```
show debug vrrp
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is for VRRP and VRRP-E. You can modify or redirect the displayed information by using the default Linux tokens (|, >).

## Examples

If you run this command and the debug parameter has already been set to debug all VRRP events, the following is displayed:

```
switch# show debug vrrp
VRRP event debugging is on
```

# show default threshold

Displays the default thresholds for environmental and alert values for Ethernet interfaces, login, Telnet security monitoring, and SFPs.

## Syntax

```
show defaults threshold [ interface type Ethernet | security | sfp ]
```

## Parameters

### interface type Ethernet

Thresholds for all Ethernet interfaces.

### security

Thresholds for login and Telnet monitoring.

### sfp

Thresholds for the following SFP types:

1 GSR

1 GSR

10 GLR

10 GER

10 GUSR

## Modes

Privileged EXEC mode

## Usage Guidelines

These thresholds can be changed by means of the **threshold-monitor** command.

## Examples

The following example illustrates default interface thresholds:

```
switch# show defaults threshold interface type Ethernet
Type: GigE-Port
+-----+-----+-----+-----+-----+-----+-----+-----+
| Area          | High Threshold | Low Threshold | Buffer | Time | | | |
| Value | Above | Below | Value | Above | Below | Value | Base |
| Action | Action | Action | Action | Action | Action |      |      |
+-----+-----+-----+-----+-----+-----+-----+-----+
| MTC           | 300 | none | none | 12 | none | none | 0 | minute |
+-----+-----+-----+-----+-----+-----+-----+-----+
| CRCAlign     | 300 | none | none | 12 | none | none | 0 | minute |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Symbol       | 5   | none | none | 0  | none | none | 0 | minute |
+-----+-----+-----+-----+-----+-----+-----+-----+
| IFG          | 100 | none | none | 5  | none | none | 0 | minute |
+-----+-----+-----+-----+-----+-----+-----+-----+
MTC - Missing Termination Character
```

The following example illustrates security thresholds:

```
sw0# show defaults threshold security
+-----+-----+-----+-----+-----+-----+-----+-----+
| Area          | High Threshold | Low Threshold | Buffer | Time | | |
| Value | Above | Below | Value | Below | Value | Base |
| Action | Action | Action | Action | Action |      |      |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Telnet        | 2 | raslog | none | 1 | none | 0 | minute |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Login         | 2 | raslog | none | 1 | none | 0 | minute |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

# show default-vlan

Displays the current default VLAN value.

## Syntax

`show default-vlan`

## Modes

Privileged EXEC mode

## Related Commands

[show running reserved-vlan, reserved-vlan](#)



# show dhcpd configuration

Displays the configured DHCP server details on the VDX 6740T.

## Syntax

```
show dhcpd configuration [rbrdgerbridge-id]
```

## Modes

Privileged EXEC mode

## Usage Guidelines

## Examples

The following example displays the dhcpd configuration.

```
device# show dhcpd configuration
subnet 20.1.1.0 netmask 255.255.255.0 {
  option routers 20.1.1.5;
  range 20.1.1.150 20.1.1.155;
  option domain-name-servers 20.1.1.150;
  option subnet-mask 255.255.255.0;
  default-lease-time 1200;
  max-lease-time 7200;
  option broadcast-address 20.1.1.255;
  option host-name dhcpclient;
}
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show dpod

Displays Dynamic Ports on Demand (DPOD) licensing.

## Syntax

```
show dpod [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged Exec mode

## Usage Guidelines

This command has no effect on Extreme VDX 8770 devices. These devices do not support the Dynamic POD feature.

Remote license operations may be performed on any remote RBridge, from any RBridge in the cluster.

## Examples

```

device# show dpod

rbridge-id: 15
48 10G ports are available in this switch
4 40G ports are available in this switch
10G Port Upgrade license is installed
40G Port Upgrade license is installed
Dynamic POD method is in use
32 10G port assignments are provisioned for use in this switch:
24 10G port assignments are provisioned by the base switch license
8 10G port assignments are provisioned by the Port Upgrade license
1 10G port is assigned to installed licenses:
1 10G port is assigned to the base switch license
0 10G ports are assigned to the Port Upgrade license
10G ports assigned to the base switch license:
15/0/12
10G ports assigned to the Port Upgrade license:
None
10G ports not assigned to a license:
15/0/1, 15/0/2, 15/0/3, 15/0/4, 15/0/5, 15/0/6, 15/0/7, 15/0/8, 15/0/9,
15/0/10
15/0/11, 15/0/13, 15/0/14, 15/0/15, 15/0/16, 15/0/17, 15/0/18, 15/0/19,
15/0/20, 15/0/21
15/0/22, 15/0/23, 15/0/24, 15/0/25, 15/0/26, 15/0/27, 15/0/28, 15/0/29,
15/0/30, 15/0/31
15/0/32, 15/0/33, 15/0/34, 15/0/35, 15/0/36, 15/0/37, 15/0/38, 15/0/39,
15/0/40, 15/0/41
15/0/42, 15/0/43, 15/0/44, 15/0/45, 15/0/46, 15/0/47, 15/0/48
31 10G license reservations are still available for use by unassigned ports
16 40G port assignments are provisioned for use in this switch:
0 40G port assignments are provisioned by the base switch license
2 40G port assignments are provisioned by the Port Upgrade license
1 40G port is assigned to installed licenses:
0 40G ports are assigned to the base switch license
1 40G ports are assigned to the Port Upgrade license
40G ports assigned to the base switch license:
None
40G ports assigned to the Port Upgrade license:
15/0/49
40G ports not assigned to a license:
15/0/50, 15/0/51, 15/0/52
3 40G license reservations are still available for use by unassigned ports

```

## Related Commands

[show running-config dpod](#)

# show diag burninerrshow

Displays the error messages that are stored in the nonvolatile storage on the slot during the POST and system verification processes.

## Syntax

```
show diag burninerrshow [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*  
Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

The error messages are updated when there is a POST failure or a systemVerification failure. To display burn-in errors from the switch:

```
switch# show diag burninerrshow rbridge-id 1

errLog for slot M2
errLog is empty for slot M2
errLog for slot S1
errLog is empty for slot S1
errLog for slot S2
errLog is empty for slot S2
errLog for slot S3
errLog is empty for slot S3
errLog for slot L4
errLog is empty for slot L4
rbridgeId 1
```

## Related Commands

[diag burninerrclear](#), [diag clearerror](#), [show diag burninstatus](#)

# show diag burninerrshowerrLog

Displays the error log messages that are stored in the nonvolatile storage on the slot during the POST and system verification processes.

## Syntax

```
show diag burninerrshowerrLog [ slot slot-id ]
```

## Parameters

**slot** *slot\_id*

Specifies the slot ID. This is mandatory for slot-based systems only.

## Modes

Privileged EXEC mode

## Examples

The error messages are updated when there is a POST failure or a systemVerification failure. To display the error log messages on the slot:

```
switch# show diag burninerrshowerrLog
Log for slot MlerrLog is empty for slot S12012/06/03-07:11:17:038992, [DIAG-5004], 0,
M1, INFO, chassis, DIAG-MANUAL4 " S1 verify: Starting run Sun Jun 3 07:11:14 PDT 2012 "Err# 0140045
0300:101:000:001:0:20: , OID:0x430c0000, iobuf.c, line: 648, comp:insmod, ltime:
2012/06/03-07:2012/06/03-07:31:02:766063, [DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " S1 verify:
TESTED stat PASSED 5 cmds in 1 runs Therm 10 Vib 2 in 0 hr 18 min 53 sec (0:18:53)"Err# 0140045
0300:101:000:001:0:20: , OID:0x430c0000, iobuf.c, lineerrLog for slot S22012/06/03-07:11:16:618653,
[DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " S2 verify: Starting run Sun Jun 3 07:11:13 PDT 2012
"Err# 0140045 0400:101:000:001:0:20: , OID:0x43100000, iobuf.c, line: 648, comp:insmod, ltime:
2012/06/03-07:2012/06/03-07:30:39:636631, [DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " S2 verify:
TESTED stat PASSED 5 cmds in 1 runs Therm 10 Vib 2 in 0 hr 18 min 58 sec (0:18:58)"Err# 0140045
0400:101:000:001:0:20: , OID:0x43100000, iobuf.c, lineerrLog for slot S32012/06/03-07:11:12:838561,
[DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " S3 verify: Starting run Sun Jun 3 07:11:09 PDT 2012
"Err# 0140045 0500:101:000:001:0:20: , OID:0x43140000, iobuf.c, line: 648, comp:insmod, ltime:
2012/06/03-07:2012/06/03-07:30:35:017964, [DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " S3 verify:
TESTED stat PASSED 5 cmds in 1 runs Therm 10 Vib 2 in 0 hr 19 min 4 sec (0:19:4)"Err# 0140045
0500:101:000:001:0:20: , OID:0x43140000, iobuf.c, line:errLog for slot L12012/06/03-07:11:18:678484,
[DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " L1 verify: Starting run Sun Jun 3 07:11:15 PDT 2012
"Err# 0140045 0700:101:000:001:0:20: , OID:0x431c0000, iobuf.c, line: 648, comp:insmod, ltime:
2012/06/03-07:2012/06/03-07:30:56:177298, [DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " L1 verify:
TESTED stat PASSED 8 cmds in 1 runs Therm 10 Vib 2 in 0 hr 18 min 44 sec (0:18:44)"Err# 0140045
0700:101:000:001:0:20: , OID:0x431c0000, iobuf.c, lineerrLog for slot L22012/06/03-07:11:18:678576,
[DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " L2 verify: Starting run Sun Jun 3 07:11:15 PDT 2012
"Err# 0140045 0800:101:000:001:0:20: , OID:0x43200000, iobuf.c, line: 648, comp:insmod, ltime:
2012/06/03-07:2012/06/03-07:30:40:774116, [DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " L2 verify:
TESTED stat PASSED 8 cmds in 1 runs Therm 10 Vib 2 in 0 hr 18 min 41 sec (0:18:41)"Err# 0140045
0800:101:000:001:0:20: , OID:0x43200000, iobuf.c, lineerrLog for slot L32012/06/03-07:11:17:097345,
[DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " L3 verify: Starting run Sun Jun 3 07:11:14 PDT 2012
"Err# 0140045 0900:101:000:001:0:20: , OID:0x43240000, iobuf.c, line: 648, comp:insmod, ltime:
2012/06/03-07:2012/06/03-07:19:29:651740, [DIAG-5046], 0, M1, ERROR, chassis, L3:portLoopbackTest
FAILED. Err -2, OID:0x43240000, diag_mercury_mm, line: 543, comp:diag, ltime:
2012/06/03-07:19:29:6516992012/06/03-07:29:52:276612, [DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 "
L3 verify: TESTED stat FAILED 8 cmds in 1 runs Therm 10 Vib 2 in 0 hr 18 min 34 sec (0:18:34)"Err#
0140045 0900:101:000:001:0:20: , OID:0x43240000, iobuf.c, lineerrLog for slot
L42012/06/03-07:11:17:385343, [DIAG-5004], 0, M1, INFO, chassis, DIAG-MANUAL4 " L4 verify: Starting run
Sun Jun 3 07:11:15 PDT 2012 "Err# 0140045 0A00:101:000:001:0:20: , OID:0x43280000, iobuf.c, line: 648,
comp:insmod, ltime:2012/06/03-07:2012/06/03-07:30:27:647391, [DIAG-5004], 0, M1, INFO, chassis, DIAG-
MANUAL4 " L4 verify: TESTED stat PASSED 8 cmds in 1 runs Therm 10 Vib 2 in 0 hr 18 min 55 sec
(0:18:55)"Err# 0140045 0A00:101:000:001:0:20: , OID:0x43280000, iobuf.c, linerbridgeId 233M4_237_233#
```

## Related Commands

[diag burninerrclear](#), [diag clearerror](#), [show diag burninstatus](#)

# show diag burninstatus

Displays the diagnostics burn-in status or system verification status stored in the nonvolatile storage memory in the switch.

## Syntax

```
show diag burninstatus [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To display the diagnostics burn-in status:

```
switch# show diag burninstatus
```

DiagID	State	Status	Run	Cmd	TotCmds	PID	Script	SlotID
1	COMPLETE_TESTED	PASS	1	8	8	23163	verify	L1
2	COMPLETE_TESTED	PASS	1	8	8	23311	verify	L2
6	COMPLETE_TESTED	PASS	1	5	5	23465	verify	S2
7	COMPLETE_TESTED	PASS	1	5	5	23618	verify	S3
8	COMPLETE_TESTED	PASS	1	5	5	23787	verify	S4
9	COMPLETE_TESTED	PASS	1	5	5	23976	verify	S5
10	COMPLETE_TESTED	PASS	1	5	5	24156	verify	S6
12	COMPLETE_TESTED	PASS	1	8	8	24388	verify	L6
14	COMPLETE_TESTED	PASS	1	8	8	24692	verify	L8

```
rbridgeId 1
```

## Related Commands

[diag burninerrclear](#), [diag clearerror](#), [show diag burninerrshow](#)

# show diag post results

Displays either the brief results or detailed information of the power-on self-test (POST) executed on the switch.

## Syntax

```
show diag post results { brief | detailed } [ rbridge-id rbridge-id ] [ slot slot-id ]
```

## Parameters

**brief | detailed**

Specifies whether the POST passed or failed (brief) or displays detailed status with the register dump when a POST fails (detailed).

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**slot** *slot\_id*

Specifies the slot ID. This is mandatory for slot-based systems only.

## Modes

Privileged EXEC mode



## Examples

To display brief POST results (whether the POST passed or failed):

```
switch# show diag post results brief slot L4

POST1:Slot L4 turboramtest PASSED (exit_status 0).
POST1:Slot L4 Script PASSED with exit_status of 0 Thu Jan 1 00:04:36 GMT 1970 took (0:0:47)
POST2:Slot L4 portloopbacktest PASSED (exit_status 0).
POST2:Slot L4 prbstest PASSED (exit_status 0).
POST2:Slot L4 Script PASSED with exit_status of 0 Thu Jan 1 00:05:52 GMT 1970 took (0:1:15)
rbridgeId 1
switch# show diag post results detailed slot S1

POST1:Slot S1 Started running Thu Jan 1 00:02:46 GMT 1970
POST1:Slot S1 Running diagclearerror
POST1:Slot S1 Running diagsetup
POST1:Slot S1 Test #1 - Running turboramtest
Running turboramtest...
:
<..cut..>
:
POST1:Slot S1 ***** Slot S1 POST Summary *****
POST1:Slot S1 Completed 1 Diagnostic test:
POST1:Slot S1 Script PASSED with exit_status of 0 Thu Jan 1 00:02:53 GMT 1970 took (0:0:7)
POST2:Slot S1 Started running Thu Jan 1 00:02:58 GMT 1970
POST2:Slot S1 Running diagclearerror
POST2:Slot S1 Test #1 - Running portloopbacktest
Running portloopbacktest...
:
<..cut..>
:
POST2: ***** Slot S1 POST Summary *****
POST2:Slot S1 Completed 2 Diagnostic test:
POST2:Slot S1 Script PASSED with exit_status of 0 Thu Jan 1 00:03:35 GMT 1970 took (0:0:37)
rbridgeId 1
```

## Related Commands

[diag post enable](#)

# show diag setcycle

Displays the current system verification test parameters.

## Syntax

```
show diag setcycle [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*  
Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To display current values used in system verification:

```
switch# show diag setcycle

CURRENT - KEYWORD      : DEFAULT
 1      - number_of_runs : 1
 2      - min_lb_mode    : 2
 1      - tbr_passes     : 1
16      - plb_nframes    : 16
 1      - pled_passes    : 1
rbridgeId 1
```

## Related Commands

[diag setcycle](#)

# show diag status

Displays the currently diagnostic test status on one or all slots in the system.

## Syntax

```
show diag status [ rbridge-id rbridge-id ] [ slot slot-id ]
```

## Command Default

If an RBridge ID is not specified, diagnostic tests for all blades in the system are displayed.

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**slot** *slot\_id*

Specifies the slot ID. This is mandatory for slot-based systems only.

## Modes

Privileged EXEC mode

## Examples

To automatically display current diagnostic status in the console:

```
switch# show diag status rbridge-id 1

Slot M2 [2]: DIAG runs 'NONE'
Slot S1 [3]: DIAG runs 'NONE'
Slot S2 [4]: DIAG runs 'NONE'
Slot S3 [5]: DIAG runs 'NONE'
Slot L4 [10]: DIAG runs 'NONE'
rbridgeId 1
```

To display the diagnostic status when POST is running on the LC or SFM using the slot ID:

```
switch# show diag status rbridge-id 233 slot L1

Slot L1 [7]:DIAG runs `turboramtest'
rbridgeId 233
switch# show diag status slot L1

Slot L1 [7]: DIAG runs `turboramtest'
rbridgeID 233
```

## Related Commands

[show diag post results](#)

# show dot1x

Displays the overall state of dot1x on the system.

## Syntax

`show dot1x`

## Modes

Privileged EXEC mode

## Examples

To display the state of dot1x on the system:

```
switch# show dot1x

802.1X Port-Based Authentication Enabled
PAE Capability:           Authenticator Only
Protocol Version:        2
Auth Server:             RADIUS
RADIUS Configuration
-----
Position:                 1
Server Address:          172.21.162.51
Port:                    1812
Secret:                  sharedsecret
Position:                 2
Server Address:          10.32.154.113
Port:                    1812
Secret:                  sharedsecret
```

# show dot1x all

Displays detailed dot1x information for all of the ports.

## Syntax

```
show dot1x all
```

## Modes

Privileged EXEC mode

## Examples

To display detailed dot1x information for all of the ports:

```
switch# show dot1x all

802.1X Port-Based Authentication Enabled
PAE Capability:          Authenticator Only
Protocol Version:      2
Auth Server:           RADIUS
802.1X info for interface te0/16
-----
Port Control:          Auto
Port Auth Status:      Unauthorized
Protocol Version:      2
ReAuthentication:      Disabled
Auth Fail Max Attempts: 0
ReAuth Max:            2
Tx Period:              30 seconds
Quiet Period:          60 seconds
Supplicant Timeout:    30 seconds
Server Timeout:        30 seconds
Re-Auth Interval:      3600 seconds
PAE State:              Connected
BE State:               Invalid
Supplicant Name:        --
Supplicant Address:    0000.0000.0000
Current Id:             1
Id From Server:        0
```

# show dot1x diagnostics interface

Displays all diagnostics information for the authenticator associated with a port.

## Syntax

```
show dot1x diagnostics interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

To display all diagnostics information for the authenticator associated with a port:

```
switch# show dot1x diagnostics interface tengigabitethernet 5/0/16
```

```
802.1X Diagnostics for interface te5/0/16
authEnterConnecting: 0
authEaplogoffWhileConnecting: 1
authEnterAuthenticating: 0
authSuccessWhileAuthenticating: 0
authTimeoutWhileAuthenticating: 0
authFailWhileAuthenticating: 0
authEapstartWhileAuthenticating: 0
authEaplogoggWhileAuthenticating: 0
authReauthsWhileAuthenticated: 0
authEapstartWhileAuthenticated: 0
authEaplogoffWhileAuthenticated: 0
BackendResponses: 0
BackendAccessChallenges: 0
BackendOtherrequestToSupplicant: 0
BackendAuthSuccess: 0
BackendAuthFails: 0
```

# show dot1x interface

Displays the state of a specified interface.

## Syntax

```
show dot1x interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **gigabitethernet** *rbridge-id/slot/port* parameter is used only on Extreme VDX 6710, Extreme VDX 8770-4, and Extreme VDX 8770-8 switches.

## Examples

To display the state of the 10-gigabit Ethernet interface 0/16:

```
switch# show dot1x interface tengigabitethernet 5/0/16
```

```
Dot1x Global Status:      Enabled
802.1X info for interface te5/0/16
-----
Port Control:             Auto
Port Auth Status:         Unauthorized
Protocol Version:         2
ReAuthentication:         Disabled
Auth Fail Max Attempts:   0
ReAuth Max:               2
Tx Period:                30 seconds
Quiet Period:             60 seconds
Supplicant Timeout:       30 seconds
Server Timeout:           30 seconds
Re-Auth Interval:         3600 seconds
PAE State:                 Connected
BE State:                  Invalid
Supplicant Name:          --
Supplicant Address:       0000.0000.0000
Current Id:                1
Id From Server:           0
```



# show dot1x session-info interface

Displays all statistical information of an established session.

## Syntax

```
show dot1x session-info interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

To display all statistical information of the established session:

```
switch# show dot1x session-info interface tengigabitethernet 0/16

802.1X Session info for te0/16
-----
User Name:                testuser
Session Time:             3 mins 34 secs
Terminate Cause:         Not terminated yet
```

# show dot1x statistics interface

Displays the statistics of a specified interface.

## Syntax

```
show dot1x statistics interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **gigabitethernet** *rbridge-id/slot/port* parameter is used only on Extreme VDX 6710, Extreme VDX 8770-4, and Extreme VDX 8770-8 switches.

## Examples

To display the statistics for the 10-gigabit Ethernet interface 22/0/16:

```
switch# show dot1x statistics interface tengigabitethernet 22/0/16

802.1X statistics for interface te22/0/16
EAPOL Frames Rx: 0 - EAPOL Frames Tx: 0
EAPOL Start Frames Rx: 0 - EAPOL Logoff Frames Rx: 0
EAP Rsp/Id Frames Rx: 2 - EAP Response Frames Rx: 10
EAP Req/Id Frames Tx: 35 - EAP Request Frames Tx: 0
Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
EAPOL Last Frame Version Rx: 0 - EAPOL Last Frame Src: 0000.0000.0000
```

# show dpod

Displays Dynamic Ports on Demand (POD) license information.

## Syntax

```
show dpod [ rbridge-id { rbridge-id | all } ]
```

## Command Default

This command is executed on the local switch.

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

The Dynamic POD feature is not supported on Extreme VDX 8770 switches.

## Examples

To display Dynamic POD assignment information:

```
device# show dpod

rbridge-id: 1
24 ports are available in this switch
1 POD license is installed
Dynamic POD method is in use
24 port assignments are provisioned for use in this switch:
16 port assignments are provisioned by the base switch license
8 port assignments are provisioned by the first POD license
* 0 more assignments are added if the second POD license is installed
21 ports are assigned to installed licenses:
16 ports are assigned to the base switch license
5 ports are assigned to the first POD license
Ports assigned to the base switch license:
Te 1/0/1, Te 1/0/10, Te 1/0/11, Te 1/0/12, Te 1/0/13, Te 1/0/14, Te 1/0/15,
Te 1/0/16, Te 1/0/17, Te 1/0/18, Te 1/0/19, Te 1/0/20, Te 1/0/21, Te 1/0/22, Te
1/0/23, Te 1/0/24
Ports assigned to the first POD license:
Te 1/0/5, Te 1/0/6, Te 1/0/7, Te 1/0/8, Te 1/0/9
Ports assigned to the second POD license:
None
Ports not assigned to a license:
Te 1/0/2, Te 1/0/3, Te 1/0/4
3 license reservations are still available for use by unassigned ports
```

## Related Commands

[dpod](#), [show running-config dpod](#)

# show dport-test

Displays the configuration, status, and results of diagnostic port (D\_Port) testing.

## Syntax

```
show dport-test { all | interface <N>gigabitethernet rbridge-id/slot/port [ detail ] | rbridge-id { rbridge-id | all } }
```

## Parameters

### all

Displays results for all interfaces tested.

### interface

Displays results for a specified TenGigabitEthernet, FortyGigabitEthernet, or HundredGigabitEthernet interface.

### detail

Specifies detailed interface information.

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

This example displays basic test results for a specified interface.

```
device# show dport-test interface Tengigabitethernet 51/0/41
D-Port Information:
=====
Interface Name:                Te 51/0/41
User Port:                    104
Remote User port:             96
Remote WWNN:                  10:00:00:27:f8:cf:49:39
Mode:                          Manual
No. of test frames:           1 Million
Test frame size:              1024 Bytes
FEC (enabled/option/active):  No/No/No
CR (enabled/option/active):   Yes/No/No
Start time:                   Thu May 28 08:30:07 2015
End time:                     Thu May 28 08:30:30 2015
Status:                        PASSED
=====
Test                           Start time    Result        EST (HH:MM:SS)  Comments
=====
Link traffic test              08:30:08     PASSED       -----        -----
=====
Roundtrip link latency: unknown
```

This example displays test results for the interface, as well as frame statistics for the port under test and any detailed errors.

```

device# show dport-test interface Te 51/0/13 detail
D-Port Information:
=====
Interface Name:                Te 51/0/13
User Port:                    76
Remote User port:             76
Remote WWNN:                  10:00:00:27:f8:cf:49:39
Mode:                         Manual
No. of test frames:          1 Million
Test frame size:             1024 Bytes
FEC (enabled/option/active): No/No/No
CR (enabled/option/active):  Yes/No/No
Start time:                  Thu May 28 08:19:49 2015
End time:                    Thu May 28 08:20:12 2015
Status:                      PASSED
=====
Test                           Start time    Result        EST(HH:MM:SS)  Comments
=====
Link traffic test              08:19:50     PASSED       -----        -----
=====
Roundtrip link latency: unknown
=====
D-Port statistics:
=====
                RX                                TX
Packets         1481857                                1481845
Bytes           1618157028                            1618143576
Unicasts        1481826                                1481814
Multicasts      31                                       31
Broadcasts      0                                       0
Errors          0                                       0
Discards        0                                       0
Overruns        0           Underruns                            0
Runts           0
Jabbers         0
CRC             0
64-byte pkts   0
Over 64-byte pkts 24
Over 127-byte pkts 7
Over 255-byte pkts 0
Over 511-byte pkts 0
Over 1023-byte pkts 1481826
Over 1518-byte pkts 0
Mbits/Sec      0.000000                            0.000000
Packet/Sec     0                                       0
Line-rate      0.00%                               0.00%
=====
Spinfab Log:
=====
Logs Unavailable

```

This example displays test results for all D\_Ports in the fabric.

```
device# show dport-test all
RBridge-Id : 51
=====
Index   Interface   State      SFP Capabilities   Test Result
=====
76      Te 51/0/13   ONLINE    ---                PASSED
104     Te 51/0/41   ONLINE    ---                PASSED
=====

RBridge-Id : 52
=====
Index   Interface   State      SFP Capabilities   Test Result
=====
76      Te 52/0/13   ONLINE    ---                PASSED
96      Te 52/0/33   ONLINE    ---                PASSED
=====
```

This example displays summary test results for all D\_Ports in a specified RBridge.

```
device# show dporttest rbridge-id 52

RBridge-Id : 52
=====
Index   Interface   State      SFP Capabilities   Test Result
=====
76      Te 52/0/13   ONLINE    ---                PASSED
=====
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show edge-loop-detection detail

Displays ELD detailed information for the entire node.

## Syntax

```
show edge-loop-detection detail [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

ELD configuration mode

## Usage Guidelines

This functionality detects Layer 2 loops only.

If no rbridge ID is specified, ELD data on the particular node is displayed.

If an rbridge ID is specified, ELD data for the node with that particular rbridge-id is displayed.

If all rbridge IDs are specified, ELD data from all the nodes in the cluster is displayed.

## Examples

```
switch(conf-if-te-119/0/1)# do show edge-loop-detection detail
Number of edge-loop-detection instances enabled: 1
  Data for Rbridge-id: 119
  Total_instances: 1
  Eld-mac: 03:05:33:65:1b:ec
  Data for interface: te0/1
  Eld-instance no. (enabled for VLANs): 1
  Priority: 128   If_status: 1
  Shutdown-vlan: 0   Vlag-master-id: 0   Age-left: 31913 mins
  Port-type : 3   pvid_frame_type: 2   Brcd-agg-type: 0
  Eld stats:      Tx      Rx
                 42      0
  Enabled for Vlan-id: 10
  Send-untagged: 0
  time-rxlimit : 0
  Vlan stats:     Tx      Rx
                 42      0
switch(conf-if-te-119/0/1)#
```



# show edge-loop-detection globals

Displays ELD global configuration values for status, disabled ports, and resource.

## Syntax

```
show edge-loop-detection globals
```

## Modes

Privileged EXEC mode

ELD configuration mode

## Usage Guidelines

The command output displays the PDU receive limit, shutdown time, and hello time.

This command detects Layer 2 loops only.

## Examples

To view the ELD global configuration values:

```
switch# show edge-loop-detection globals
```

```
Edge-loop-detection global configuration values are as below:  
PDU receive limit (packets):    1  
Shutdown-time (minutes):       0  
Hello-time (msec):             1000
```

# show edge-loop-detection interface

Displays ELD configuration settings and status for a specific interface and to view the number of packets received and transmitted.

## Syntax

```
show edge-loop-detection interface interface-type interface-id
```

```
show edge-loop-detection interface { <N>gigabitethernet rbridge-id/slot/port | port-channel number }
```

## Parameters

*interface-type*

Specifies an interface type.

*interface-id*

Specifies an interface ID.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies the interface is a port-channel. Valid values range from 1 through 6144.

## Modes

Privileged EXEC mode

ELD configuration mode

## Usage Guidelines

This functionality detects Layer 2 loops only.

## Examples

To view the ELD status for a specific interface:

```
switch(conf-if-te-7/0/5)# do show edge-loop-detection interface tengigabitethernet 7/0/5
```

```
Number of eld instances: 1
Enabled on VLANs:      100
Priority:              128
Interface status:     UP
Auto enable in:       Never
Packet Statistics:
vlan      sent      rcvd
100      100         0
switch(conf-if-te-7/0/5)# do show edge-loop-detection rbridge-id 7
```

```
Number of edge-loop-detection instances enabled: 1
Interface: 7/0/5
```

```
-----
Enabled on VLANs: 100
Priority:         128
Interface status: UP
Auto enable in:  Never
```

# show edge-loop-detection rbridge-id

Displays ELD status information for a specific RBridge, including the ports that ELD has disabled..

## Syntax

```
show edge-loop-detection rbridge-id { rbridge-id | all }
```

## Parameters

- rbridge-id**  
Specifies an RBridge or all RBridges.
- rbridge-id*  
Specifies an RBridge ID.
- all**  
Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

This functionality detects Layer 2 loops only.

For each interface on which ELD is enabled, this command shows the enabled VLANs, the ELD port priority, the up/down status of the interface, and time to the next automatic port re-enable. The command includes display of disabled interfaces.

If a single rbridge ID is specified, ELD data for the node with that particular rbridge-id is displayed.

If all rbridges are specified, ELD data from all the nodes in the cluster is displayed.

## Examples

To view the ELD status:

```
switch# show edge-loop-detection rbridge-id 7

Number of edge-loop-detection instances enabled: 1
Interface: 7/0/5
-----
Enabled on VLANs: 100
Priority:         128
Interface status: UP
Auto enable in:  Never
```

# show environment fan

Displays fan status information.

## Syntax

```
show environment fan [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show environment fan** command displays the following information:

Output field	Description
OK	Fan is functioning correctly at the displayed speed (RPM).
absent	Fan is not present.
below minimum	Fan is present but rotating too slowly or stopped.
above maximum	Fan is rotating too quickly.
unknown	Unknown fan unit installed.
faulty	Fan has exceeded hardware tolerance and has stopped. In this case, the last known fan speed is displayed.
Airflow direction	Port side intake or Port side exhaust. This value is not applicable to modular chassis.
speed	Fan RPM.

## Examples

The following example displays fan status information:

```
device# show environment fan

Fan 1 is Ok, speed is 2057 RPM
Fan 2 is Ok, speed is 2009 RPM
```

# show environment history

Displays the field-replaceable unit (FRU) history log.

## Syntax

```
show environment history [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

The history log records insertion and removal events for field-replaceable units (FRUs), such as blades, power supplies, fans, and world wide name (WWN) or chassis ID (CID) cards. The type of FRU supported depends on the hardware platform.

## Command Output

The **show environment history** command displays the following information:

Output field	Description
Object type	On standalone platforms: FAN, POWER SUPPLY, WWN (WWN card), or UNKNOWN. On modular platforms: CHASSIS, CID (chassis ID card), FAN, POWER SUPPLY, SW BLADE (port blade), MM[1-2] (management module), SFM (switch fabric module), LC[1-8] (line card) or UNKNOWN.
Object number	Displays the slot number for blades. Displays the unit number for all other object types.
Event type	Displays Inserted, Removed, or Invalid.
Time of the event	Displays the date in the following format: Day Month dd hh:mm:ss yyyy.
Factory Part Number	Displays the part number (xx-yyyyyyy-zz) or Not available.
Factory Serial Number	Displays the FRU serial number (xxxxxxxxxxx) or Not available.

## Examples

The following example displays the FRU history on a device.

```
device# show environment history

CID Unit 2          Inserted at Tue Sep  6 22:40:27 2011
Factory Part Number: 40-1000592-01
Factory Serial Number: BVW0311G00K
SFM Slot 3         Inserted at Tue Sep  6 22:41:47 2011
Factory Part Number: 60-1002180-05
Factory Serial Number: BVU0321G00N
LC Slot 9          Inserted at Tue Sep  6 22:41:48 2011
Factory Part Number: 60-1002181-07
Factory Serial Number: BVV0326G019
LC Slot 10         Inserted at Tue Sep  6 22:41:50 2011
Factory Part Number: 40-100522-02
Factory Serial Number: BSX0312G01F
MM Slot 1          Inserted at Tue Sep  6 22:41:50 2011
Factory Part Number: 60-1002179-07
Factory Serial Number: BVT0329G00B
SFM Slot 4         Inserted at Wed Sep  7 00:01:44 2011
Factory Part Number: 60-1002180-06
Factory Serial Number: BVU0329G00B
LC Slot 10         Removed at Mon Sep 12 19:04:58 2011
Factory Part Number: 40-100522-02
Factory Serial Number: BSX0312G01F
LC Slot 10         Inserted at Mon Sep 12 19:12:21 2011
Factory Part Number: 40-100522-02
Factory Serial Number: BSX0312G01F
LC Slot 1          Inserted at Mon Sep 12 19:19:52 2011
Factory Part Number: 40-100522-02
Factory Serial Number: BSX0312G00B
(Output truncated)
```

```
device# show environment history
POWER SUPPLY Unit 1 Inserted at Sun Jul 12 21:59:17 2015
Factory Part Number:
Factory Serial Number:

POWER SUPPLY Unit 2 Inserted at Sun Jul 12 21:59:17 2015
Factory Part Number:
Factory Serial Number:
```

```
FAN Unit 1 Inserted at Sun Jul 12 21:59:17 2015
Factory Part Number: 60-1003113-03
Factory Serial Number: DUX0343K00A
```

```
FAN Unit 2 Inserted at Sun Jul 12 22:02:40 2015
Factory Part Number: 60-1003113-03
Factory Serial Number: DUX0343K00B
```

```
FAN Unit 3 Inserted at Sun Jul 12 22:02:41 2015
Factory Part Number: 60-1003113-03
Factory Serial Number: DUX0343K00C
```

```
SFM Slot S1 Inserted at Sun Jul 12 22:02:41 2015
Factory Part Number: 40-0000155-xx
Factory Serial Number: LU0000000000
```

```
LC Slot L4 Inserted at Sun Jul 12 22:02:41 2015
Factory Part Number: 50-1002179-14
Factory Serial Number: BVT0307H00F
```

```
CID Unit 1 Inserted at Sun Jul 12 22:02:43 2015
Factory Part Number: 60-1003219-01
Factory Serial Number: BWF0415K00D
```

```
MM Slot M1 Inserted at Sun Jul 12 22:02:43 2015
Factory Part Number: 60-1003051-02
Factory Serial Number: DCR0341K006
```

show environment history

(Output truncated)



# show environment power

Displays the type and current status of the switch power supply.

## Syntax

```
show environment power [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show environment power** command displays the following information:

Output field	Description
OK	Power supply is functioning correctly.
absent	Power supply is not present.
unknown	Unknown power supply unit is installed.
predicting failure	Power supply is present but predicting failure. Replace the power supply as soon as possible.
faulty	Power supply is present but faulty (no power cable, power switch turned off, fuse blown, or other internal error).
Airflow	Direction of fan air flow (not applicable to modular chassis).

## Examples

The following example displays the power supply status.

```
device# show environment power

Power Supply #1 is OK
LPCS      F@ 11/01/18 type: AC V165.2 3000W
Power Supply #2 is OK
LPCS      F@ 11/01/18 type: AC V165.2 3000W
Power Supply #3 is absent
Power Supply #4 is absent
```

# show environment sensor

Displays the environment sensor status.

## Syntax

```
show environment sensor [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

The command output displays the current temperature, fan, and power supply status readings from sensors located on the switch. For an explanation of power supply status values, refer to the **show environment power** topic.

## Examples

The following example displays sensor readings on the device:

```
device# show environment sensor

sensor 1: (Temperature ) is Ok, value is 36 C
sensor 2: (Temperature ) is Ok, value is 40 C
sensor 3: (Temperature ) is Ok, value is 32 C
sensor 4: (Fan          ) is Absent
sensor 5: (Fan          ) is Ok, speed is 7345 RPM
sensor 6: (Power Supply) is Absent
sensor 7: (Power Supply) is Ok
```

# show environment temp

Displays the sensor states and temperatures.

## Syntax

```
show environment temp [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show environment temp** command displays the following information:

Output field	Description
Sensor ID	Displays the sensor ID.
Sensor state	Displays OK, Above maximum, or Absent.
Temperature	Display the temperature in Centigrade and Fahrenheit.

## Examples

The following example displays temperature readings on a standalone device.

```
device# show environment temp
```

```
Sensor  State          Centigrade    Fahrenheit
  ID
=====
  1     Ok              36            96
  2     Ok              40           104
  3     Ok              32            89
```

show environment temp

The following example displays temperature readings on a modular chassis.

```
device# show environment temp
Sensor Slot State Centigrade Fahrenheit
ID
=====
 1      1   Ok         31         87
 2      1   Ok         53        127
 3      1   Ok         52        125
 4      1   Ok         37         98
 5      2   Ok         32         89
 6      2   Ok         54        129
 7      2   Ok         52        125
 8      2   Ok         39        102
 9      7  Absent
10      8  Absent
11      9  Absent
```

# show event-handler activations

Displays operational data of activated event-handlers.

## Syntax

```
show event-handler activations [ rbridge-id rbridge-ids | all ]
```

## Parameters

### rbridge-id

Specifies one or more RBridges.

### *rbridge-ids*

Specifies an RBridge, multiple RBridges, or a range of RBridges. Any of the following formats are acceptable:

- **rbridge-id 1**
- **rbridge-id 1,2,5**
- **rbridge-id 1-5**
- **rbridge-id 1,3-5**

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

To display event-handler statistics for the local RBridge, use the **show event-handler activations** form of this command.

To display event-handler statistics for one or more remote RBridges, use a **show event-handler activations rbridge-id** form of this command.

## Command Output

The **show event-handler activations** command displays the following information:

Output field	Description
Event-handler	Displays the event-handler name.
Last Trigger Activation Time	Displays the time of the last trigger activation. If no trigger was activated, displays "Never".
Total Trigger Activations	Displays the total number of trigger activations.
Last Action Completion Time	Displays the completion time of the last event-handler action run. If no event-handler action ran, displays "Never".
Last Action Completion Status. Exit Code =	Displays the status of the last completed event-handler action. If the Python script assigns exit codes, such codes are displayed here. An exit code of 0 indicates one of the following: <ul style="list-style-type: none"> <li>• No code was assigned to this condition.</li> </ul>

Output field	Description
	<ul style="list-style-type: none"> <li>The script author assigned 0 to a specified condition.</li> </ul>
Total Action Completions	Displays the number of completed event-handler actions.

## Examples

The following example displays event-handler operational data for RBridge 2.

```
device# show event-handler activations rb 2

Event-handler : evh1
Last Trigger Activation Time: 2015-04-30 17:28:12
Total Trigger Activations: 25
Last Action Completion Time: 2015-04-30 17:28:57
Last Action Completion Status: Exit Code = 0
Total Action Completions: 25

Event-handler : evh2
Last Trigger Activation Time: 2015-04-28 22:02:51
Total Trigger Activations: 8
Last Action Completion Time: 2015-04-28 22:02:58
Last Action Completion Status: Exit Code = 0
Total Action Completions: 8
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

# show fabric all

Displays the Extreme VCS Fabric membership information.

## Syntax

```
show fabric all
```

## Modes

Privileged EXEC mode

## Usage Guidelines

If the switch is initializing or is disabled, the message "Local Switch disabled or fabric is re-building" is displayed. If the fabric is reconfiguring, some or all switches may not be displayed.

## Command Output

The **show fabric all** command displays the following information:

Output field	Description
Rbridge-id	The RBridge ID of the switch.
WWN	The switch World Wide Name.
IP Address	The switch Ethernet IP address.
Name The switch symbolic name.	An arrow (>) indicates the principal switch. An asterisk (*) indicates the switch on which the command is entered.

## Examples

```
switch# show fabric all
```

```
Config Mode: Logical Chassis
```

Rbridge-id	WWN	IP Address	Name
1	10:00:00:05:1E:CD:44:6A	10.17.87.144	"RB1"
2	10:00:00:05:1E:CD:42:6A	10.17.87.145	"RB2"*
3	10:00:00:05:1E:CD:55:6A	10.17.87.155	"RB3"
4	10:00:00:05:1E:CD:42:EA	10.17.87.156	"RB4"
5	10:00:00:05:1E:CD:52:6A	10.17.87.157	"RB5"
6	10:00:00:05:1E:CD:53:6A	10.17.87.158	"RB6"
10	10:00:00:05:33:13:6A:BE	10.17.87.169	"RB10"
11	10:00:00:05:1E:CD:38:6A	10.17.86.240	"RB11"
12	10:00:00:05:1E:CD:3F:EA	10.17.86.241	>"RB12-a"

```
The Fabric has 9 Rbridge(s)
```

The angle bracket (>), as shown with RBridge ID 12-a, indicates the principal switch. The asterisk (\*) indicates the switch on which the command was entered.

show fabric all

## Related Commands

[show vcs](#)



# show fabric ecmp group

Displays the ECMP group information for the fabric RBridge IDs.

## Syntax

```
show fabric ecmp group [ dest-rbridge number | src-rbridge number ]
```

## Parameters

**dest-rbridge** *number*

Restricts the display output to the designated destination RBridge ID.

**src-rbridge** *number*

Restricts the display output to the designated source RBridge ID.

## Modes

Privileged EXEC mode

## Examples

Typical command output for the **show fabric ecmp group** command.

```
switch# show fabric ecmp group
```

```
Source RBridge-Id: 1
Total Path Count: 1
```

ECMP Grp	Dst RB-ID	Out Index	Out Interface	Nbr Index	Nbr Interface	Hops	BW	Trunk
1	2	17	Te 1/8/18	25	Te 22/0/18	2	20G	Yes

```
Source
Range
-----
1/8/1-48
1/6/1-12
```

```
Source RBridge-Id: 1
Total Path Count: 1
```

ECMP Grp	Dst RB-ID	Out Index	Out Interface	Nbr Index	Nbr Interface	Hops	BW	Trunk
1	3	17	Te 1/8/18	25	Te 22/0/18	3	20G	Yes

```
Source
Range
-----
1/8/1-48
1/6/1-12
```

```
Source RBridge-Id: 1
Total Path Count: 1
```

ECMP Grp	Dst RB-ID	Out Index	Out Interface	Nbr Index	Nbr Interface	Hops	BW	Trunk
1	22	17	Te 1/8/18	25	Te 22/0/18	1	20G	Yes

```
Source
Range
-----
1/8/1-48
1/6/1-12
```

```
Source RBridge-Id: 1
Total Path Count: 1
```

ECMP Grp	Dst RB-ID	Out Index	Out Interface	Nbr Index	Nbr Interface	Hops	BW	Trunk
1	160	17	Te 1/8/18	25	Te 22/0/18	2	20G	Yes

```
Source
Range
-----
1/8/1-48
1/6/1-12
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show fabric ecmp load-balance

Displays the current configuration of hash field selection and hash swap.

## Syntax

```
show fabric ecmp load-balance [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge.

This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.

## Examples

Some typical outputs of this command:

```
switch# show fabric ecmp load-balance

Fabric Ecmp Load Balance Information
-----
Rbridge-Id           : 2
Ecmp-Load-Balance Flavor : Destination MAC address and VID based load balancing
Ecmp-Load-Balance HashSwap : 0x4
switch# show fabric ecmp load-balance rbridge-id 2

Fabric Ecmp Load Balance Information
-----
Rbridge-Id           : 2
Ecmp-Load-Balance Flavor : Destination MAC address and VID based load balancing
Ecmp-Load-Balance HashSwap : 0x4
```

## Related Commands

[show fabric isl](#), [show fabric trunk](#)

# show fabric isl

Displays Inter-Switch Link (ISL) information in the fabric.

## Syntax

```
show fabric isl [ rbridge-id | all ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all switches in the fabric.

## Modes

Privileged EXEC mode

## Command Output

The **show fabric isl** command displays the following information:

Output field	Description
Rbridge-id	RBridge-id of the switch. Valid values range from 1 through 239.
#ISLs	Number of ISLs connected.
Src Index	Source index of the local RBridge.
Src Interface	Source interface of the local RBridge in the format "local-rbridge-id/slot/port". If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
Nbr Index	Neighbor Index of the ISL connected from local port. If the link is segmented and the NBR rbridge details are unavailable, "?" displays in this field.
Nbr Interface	Neighbor interface of the ISL connected from the local RBridge in the format "nbr-rbridgeid/slot/port". If the ISL is not completely up, this field will be displayed as "?/?/?".
Nbr-WWN	Neighbor WWN of the switch. If the ISL is segmented and the neighbor RBridge details are not available, then "?:?:?:?:?:?:?:?:?:?" displays in this field.
BW	Bandwidth of the traffic.
Trunk	Displays "Yes" if trunk is enabled in the ISL.
Nbr-Name	Neighbor switch name.

## Examples

This example displays Inter-Switch Link (ISL) information in the fabric.

```
switch# show fabric isl

Rbridge-id: 76 #ISLs: 2
Src      Src      Nbr      Nbr
Index   Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
  2      Te 1/0/2      56      Te 21/0/56     10:00:00:05:33:13:5F:BE 60G   Yes   "Edget12r1_1_21"
  7      Te 1/0/7      55      Te 22/0/55     10:00:00:05:33:40:31:93 40G   Yes   "Edget12r12_22"
```

This example displays ISL information and includes a segmented link.

```
switch# show fabric isl

Rbridge-id: 2 #ISLs: 4
Src      Src      Nbr      Nbr
Index   Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
  1      Te 2/0/1      1       Te 3/0/1       10:00:00:05:1E:CD:7A:7A 10G   Yes   "sw0"
  2      Te 2/0/2      ?       Te ?/?/?       ??:?:?:?:?:?:?:?:?:?:? (segmented - incompatible)
 26      Te 2/0/26     56      Te 25/0/56     10:00:00:05:33:40:2F:C9 60G   Yes   "Edget12r31_25"
 34      Te 2/0/34     58      Te 26/0/58     10:00:00:05:33:41:1E:B7 40G   Yes   "Edget12r32_26"
```

This example displays ISL information and includes an Fi port.

```
switch# show fabric isl

Rbridge-id: 66 #ISLs: 5
Src      Src      Nbr      Nbr
Index   Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
  5      Te 66/0/5      1       Te 65/0/1       10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
 15      Te 66/0/15     2       Te 65/0/2       10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
 25      Te 66/0/25     3       Te 65/0/3       10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
 35      Te 66/0/35     4       Te 65/0/4       10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
124      Fi 66/0/4      7       Fi 2/-/-       50:00:51:E4:44:40:0E:04 32G   Yes   "fcr_fd_2"
```

This ISL output example includes breakout information for switches on which breakout mode is configured.

```
switch# show fabric isl
Rbridge-id: 66 #ISLs: 5
Src Src Nbr Nbr
Index Interface Index Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
  5   Te 66/0/5:1  1   Te 65/0/1     10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
 15   Te 66/0/15  2   Te 65/0/2:1  10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
 25   Te 66/0/25  3   Te 65/0/3     10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
 35   Te 66/0/35:1 4   Te 65/0/4:4  10:00:00:05:33:5F:EA:A4 10G   Yes   "sw0"
124   Fi 66/0/4    7   Fi 2/-/-     50:00:51:E4:44:40:0E:04 32G   Yes   "fcr_fd_2"
```

This example displays ISL details and includes the breakout index of the interface (for a non-trunked port) in normal mode.

```
switch# sh fab isl
Rbridge-id: 1 #ISLs: 4
Src      Src      Nbr      Nbr
Index   Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
 30      Te 1/0/27     1       Te 48/0/49:1   XX:XX:XX:XX:XX:XX:XX:XX 10G   Yes   "sw0"
 32      Te 1/0/29     2       Te 48/0/49:2   XX:XX:XX:XX:XX:XX:XX:XX 10G   Yes   "sw0"
 34      Te 1/0/31     3       Te 48/0/49:3   XX:XX:XX:XX:XX:XX:XX:XX 10G   Yes   "sw0"
 36      Te 1/0/33     4       Te 48/0/49:4   XX:XX:XX:XX:XX:XX:XX:XX 10G   Yes   "sw0"
```

This example displays ISL details and includes the breakout index of the interface (for a trunked port) in normal mode.

```
switch# sh fab isl
Rbridge-id: 1 #ISLs: 1
  Src      Src      Nbr      Nbr
Index     Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
  5      Te 1/0/50:1    1       Te 48/0/49:1    10:00:00:05:33:E5:C1:8F  40G  Yes   "sw0"
```

This example displays ISL details in mixed mode, but no breakout index is displayed for switches running Network OS versions earlier than v4.1.0.

Output shown is for Network OS 4.0.0 for a non-trunked port in mixed mode.

```
sw0# sh fab isl
Rbridge-id: 1 #ISLs: 4
  Src      Src      Nbr      Nbr
Index     Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
 30      Te 1/0/27      1       Te 48/0/49      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
 32      Te 1/0/29      1       Te 48/0/49      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
 34      Te 1/0/31      1       Te 48/0/49      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
 36      Te 1/0/33      1       Te 48/0/49      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
```

Output shown is for Network OS 4.0.0 for a trunked port in mixed mode.

```
sw0# sh fab isl
Rbridge-id: 1 #ISLs: 1
  Src      Src      Nbr      Nbr
Index     Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
  2      Te 1/0/50      1       Te 48/0/49      10:00:00:05:33:E5:C1:8F  40G  Yes   "sw0"
```

Output shown is for Network OS 4.1.0 for a non-trunked port in mixed mode.

```
sw0# sh fab isl
Rbridge-id: 48 #ISLs: 4
  Src      Src      Nbr      Nbr
Index     Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
  1      Te 48/0/49:1    30      Te 1/0/25      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
  1      Te 48/0/49:2    32      Te 1/0/27      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
  1      Te 48/0/49:3    34      Te 1/0/29      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
  1      Te 48/0/49:4    36      Te 1/0/31      XX:XX:XX:XX:XX:XX:XX  10G  Yes   "sw0"
```

Output shown is for Network OS 4.1.0 for a trunked port in mixed mode.

```
sw0# sh fab isl
Rbridge-id: 1 #ISLs: 1
  Src      Src      Nbr      Nbr
Index     Interface  Index   Interface      Nbr-WWN          BW   Trunk  Nbr-Name
-----
 30      Te 1/0/50:1    1       Te 48/0/49      10:00:00:05:33:E5:C1:8F  40G  Yes   "sw0"
```

# show fabric islports

Displays information for all Inter-Switch Link (ISL) ports in the switch.

## Syntax

```
show fabric islports [ rbridge-id rbridge-id ] [ all ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies the RBridge ID on which the ISL ports are displayed.

all

Specifies that information for all RBridges in the fabric are displayed.

## Modes

Privileged EXEC mode

## Usage Guidelines

If this command is executed without operands, it displays the ISL information of the local RBridge.

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge.

This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.

## Command Output

The **show fabric islports** command displays the following information:

Output field	Description
Name	Switch name.
Type	Switch model and revision number.
State	Switch state. The valid values are Online, Offline, Testing, or Faulty.
Role	Switch role. The valid values are Principal, Subordinate, or Disabled.
VCS	Id VCS ID. Valid values range from 1 through 8192.
Config Mode	Extreme VCS Fabric mode. The valid values are Local-Only or Distributed.
Rbridge-id	RBridge ID of the switch. Valid values range from 1 through 239.
WWN	Switch world wide name (WWN).
FCF MAC	Mac address
Index	A number between 0 and the maximum number of supported ports on the platform. The port index identifies the port number relative to the switch.
Interface	Interface of the local RBridge in the format "local-rbridge-id/slot/port".

Output field	Description
State	Port state information: <ul style="list-style-type: none"> <li>Up—if the ISL is connected and the link is up.</li> <li>Down—no ISL is connected.</li> </ul>
Operational State	ISL state information: <ul style="list-style-type: none"> <li>ISL Fabric port—displays the world wide name (WWN) and name of the attached switch.</li> <li>(Trunk Primary) —the port is the master port in a group of trunking ports.</li> <li>(Trunk port, primary is rbridge-id/slot/port) —the port is configured as a trunking port; the primary port is rbridge-id/slot/port.</li> <li>(upstream)—the ISL is an upstream path toward the principal switch of the fabric.</li> <li>(downstream)—the ISL is a downstream path away from the principal switch of the fabric.</li> <li>Segmented, (Reason Code)—the ISL has been segmented due to the given reason code.</li> <li>Down, (Reason Code)—the ISL is down due to the given reason code.</li> </ul>

In this command, ISL segmentation is denoted as "ISL segmented, (segmentation reason)". The segmentation reason could be any of those listed below.

**TABLE 15** Segmentation reasons

Number	Segmentation	Explanation
1	RBridge ID Overlap	When a new node joins the cluster with the RBridge-id already existing in the cluster.
2	ESC mismatch, Unknown	Only on one side of the ISL, the actual ESC mismatch reason code will be displayed. On the other end, Unknown will be displayed.
3	ESC mismatch, Config Mode	Extreme VCS Fabric mode is different on both switches.
4	ESC mismatch, Distributed Config DB	The DCM Configuration DB is different on both the ends of ISL.
5	ESC mismatch, Extreme VCS Fabric License	Extreme VCS Fabric license is enabled on one end and disabled on the other side.
6	ESC mismatch, Fabric Distribution Service	FDS state is different on both the ends.
7	zone conflict	Zone configuration is not same on both the switches.
8	ISL Disabled	This port is disabled to be an ISL.
9	ISL Isolated	If BF or DIA is rejected then the port will get isolated.
10	LD inc om pat ability	ECP rejected or retries exceeded.
11	FDS Zone Conflict	FDS configuration caused zone conflict.
12	ESC mismatch, VCS Virtual Fabric Mode Conflict	Virtual Fabric mode is enabled on one end and disabled on the other end.

In this command, disabled ports are denoted as "Down (Disabled reason)". The disabled reason could be any of those listed below.



**TABLE 16** Disabled port reasons

Number	Disabled Reasons	Explanation
1	Admin	Port is persistently disabled.
2	Protocol Incomplete	The ISL couldn't complete the link protocol. Stuck in G_PORT state.
3	RBridge ID Overlap	Two nodes in the cluster requested for same RBridge ID.
4	Long distance inc om pat ability	Long distance configuration doesn't match.
5	ELP retries exceeded	Max ELP retries exceeded but no response from the other end.
6	Zone conflict	Zoning configuration overlaps.
7	ESC Config Mode Conflict	The link has been segmented due to different Extreme VCS Fabric mode.
8	ESC NOS incompatible	Other end does not support Extreme VCS Fabric technology.
9	ESC Distributed Config DB Conflict	DCM Configuration DB conflict.
10	ESC VCS Fabric License Conflict	Extreme VCS Fabric license is not enabled on one end.
11	No VCS Fabric License	Extreme VCS Fabric license is not enabled. For more than 2 nodes in the cluster VCS fabric license is mandatory.
12	ESC Fabric Distribution Service Conflict	FDS state is different.
13	Zoning FDS Configuration Conflict	FDS zoning configuration is different.

## Examples

To display information for all ISL ports in the core switch:

```
device# show fabric islports
Name:          sw0
Type:          96.5
State:         Online
Role:          Fabric Principal
VCS Id:        1
Config Mode: Local-Only
Rbridge-id:    23
WWN:           10:00:00:05:33:d1:3a:ac
FCF MAC:       00:05:33:d1:3a:ac
Index  Interface      State  Operational State
=====
  0    Fi 23/0/1        Up     ISL (protocol incomplete)
  1    Fi 23/0/2        Up     F-Port
  2    Fi 23/0/3        Up     F-Port
  3    Fi 23/0/4        Up     F-Port
  4    Fi 23/0/5        Up     Loopback-> Fi 23/0/5
  5    Fi 23/0/6        Down
  6    Fi 23/0/7        Down
  7    Fi 23/0/8        Down
  8    Te 23/0/1        Up     ISL segmented, (ESC mismatch, Config Mode) (Trunk Primary)
  9    Te 23/0/2        Down
 10   Te 23/0/3        Down
 11   Te 23/0/4        Down
 12   Te 23/0/5        Down
 13   Te 23/0/6        Down
 14   Te 23/0/7        Down
 15   Te 23/0/8        Down
 16   Te 23/0/9        Down
 17   Te 23/0/10       Down
 18   Te 23/0/11       Down
 19   Te 23/0/12       Down
 20   Te 23/0/13       Down
 21   Te 23/0/14       Down
 22   Te 23/0/15       Down
 23   Te 23/0/16       Down
 24   Te 23/0/17       Down
 25   Te 23/0/18       Down
 26   Te 23/0/19       Down
 27   Te 23/0/20       Down
 28   Te 23/0/21       Down
 29   Te 23/0/22       Down
 30   Te 23/0/23       Down
 31   Te 23/0/24       Down
```

## History

Release version	Command history
6.0.1a	This command was modified to remove reference to "standalone" mode.

# show fabric login-policy

Displays the Duplicate WWN login mode of local switches in a fabric.

## Syntax

```
show fabric login-policy [ rbridge-id { rbridge-id | all }
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the login policy on the principal node.

```
sw0# show fabric login-policy

=====
Switch Login Policy
=====
Switch Name: sw0
Rbridge-Id: 2
DuplicateWWN Login Mode: old-login.(Default)
```

The following example displays the login policy for a specified RBridge.

```
sw0# show fabric login-policy rbridge-id 51

=====
Switch Login Policy:
=====
Switch Name: sw0
Rbridge-Id: 51
DuplicateWWN Login Mode:new-login.
```

The following example displays the login policy for all RBridges in the fabric.

```
sw0# show fabric login-policy all

=====
Switch Login Policy:
=====
Switch Name: sw0
Rbridge-Id: 51
DuplicateWWN Login Mode:new-login.

Switch Name: sw0
Rbridge-Id: 52
DuplicateWWN Login Mode:old-login.(Default).
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show fabric port-channel

Displays the fabric VLAG load-balance information.

## Syntax

```
show fabric port-channel [ port-channel-id | load-balance ]
```

## Parameters

*port-channel-id*

Displays the information for the port channel ID.

**load-balance**

Displays the load balance information.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.

## Examples

```
switch# show fabric port-channel 10 load-balance
Fabric Vlag Load-Balance Information
-----
Port-channel id      : 10
Load-Balance Flavor : Source and Destination MAC address and VID based load balancing
```

## Related Commands

[show fabric isl](#), [show fabric trunk](#)

# show fabric route linkinfo

Displays the RBridge route link information connected in the fabric.

## Syntax

```
show fabric route linkinfo
```

## Modes

Privileged EXEC mode

## Usage Guidelines

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge.

The output displays the link information, which includes the breakout index of the interface if breakout mode is configured in the source or neighbor interface.

When a fabric is running normal mode (switches are running Network OS 4.1.0 or later), CLI output displays the QSFP breakout index for all switches in the fabric.

When a fabric is running mixed mode (switches are running Network OS 4.1.0 or earlier), the QSFP breakout index is not displayed in the output on switches running Network OS versions earlier than v4.1.0.

## Command Output

The **show fabric route linkinfo** command displays the following information:

Output field	Description
Rbridge-id	ID of the RBridge. Valid values range from 1 through 239.
Reachable	Indicates whether the RBridge can be reached. Displays "Yes" if it is reachable, otherwise displays "No".
Version	Displays the version.
No. of links	The number of ISLs connected to the neighbor switches.
Link #	A sequence number for links on the RBridge.
Src Index	E_Port interface on the local switch. This value is typically equal to the Index field reported in the switchShow command.
Src Interface	Source interface of the local RBridge in the format "local-rbridge-id/slot/port". If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
Nbr Index	E_Port interface on the remote switch. This value is typically equal to the index field reported in the switchShow command. If the link is segmented and the NBR rbridge details are unavailable, "?" displays in this field.
Nbr Interface	Neighbor interface of the ISL connected from the local RBridge in the format "nbr-rbridgeid/slot/port". If the ISL is not completely up, this field will be displayed as "?/?/?".
Link-Cost	The cost of reaching the destination domain.
Link-type	The type of link.
Trunk	Displays "Yes" if trunk is enabled in the ISL, otherwise displays "No".

## Examples

To display link information for the fabric:

```
switch# show fabric route linkinfo

Rbridge-id: 1
=====
Reachable:   Yes
Version:     1
No. of Links: 2
Link#      Src      Src      Nbr      Nbr
           Index    Interface Index    Interface Link-Cost Link-Type   Trunk
-----
1          1          Fi 1/-/- 128      Fi 2/-/- 10000    Pt_Pt
2          159       Fi 1/-/- 128      Fi 160/-/- 10000    Pt_Pt
Rbridge-id: 2
=====
Reachable:   Yes
Version:     1
No. of Links: 2
Link#      Src      Src      Nbr      Nbr
           Index    Interface Index    Interface Link-Cost Link-Type   Trunk
-----
1          129       Fi 2/-/- 49        Fi 66/0/1 10000    Pt_Pt
2          128       Fi 2/-/- 1         Fi 1/-/- 10000    Pt_Pt
Rbridge-id: 65
=====
Reachable:   Yes
Version:     1
No. of Links: 2
Link#      Src      Src      Nbr      Nbr
           Index    Interface Index    Interface Link-Cost Link-Type   Trunk
-----
1          2         Te 65/0/2 2         Te 66/0/2 500      Pt_Pt Ethernet Yes
2          44       Te 65/0/44 20        Te 66/0/20 500      Pt_Pt Ethernet Yes
Rbridge-id: 66
=====
Reachable:   Yes
Version:     1
No. of Links: 4
Link#      Src      Src      Nbr      Nbr
           Index    Interface Index    Interface Link-Cost Link-Type   Trunk
-----
1          2         Te 66/0/2 2         Te 65/0/2 500      Pt_Pt Ethernet Yes
2          20        Te 66/0/20 44        Te 65/0/44 500      Pt_Pt Ethernet Yes
3          49        Fi 66/0/1 129       Fi 2/-/- 500      Pt_Pt      Yes
4          54        Fi 66/0/6 129       Fi 160/-/- 500      Pt_Pt      Yes
Rbridge-id: 160
=====
Reachable:   Yes
Version:     1
No. of Links: 2
Link#      Src      Src      Nbr      Nbr
           Index    Interface Index    Interface Link-Cost Link-Type   Trunk
-----
1          129       Fi 160/-/- 54        Fi 66/0/6 10000    Pt_Pt
2          128       Fi 160/-/- 159       Fi 1/-/- 10000    Pt_Pt
```

This example displays link linformation and includes the breakout index of the interface in normal mode.

```
sw0# show fabric route linkinfo
Rbridge-id: 1
=====
Reachable:   Yes
Version:     1
No. of Links: 4
Link#   Src      Src      Nbr      Nbr
        Index   Interface Index   Interface Link-Cost  Link-Type  Trunk
-----
1       28       Te 1/0/49:1  0       Te 48/0/49:1  500       Pt_Pt Ethernet
2       30       Te 1/0/49:2  1       Te 48/0/49:2  500       Pt_Pt Ethernet
3       32       Te 1/0/49:3  2       Te 48/0/49:3  500       Pt_Pt Ethernet
4       34       Te 1/0/49:4  3       Te 48/0/49:4  500       Pt_Pt Ethernet
Rbridge-id: 48
=====
Reachable:   Yes
Version:     1
No. of Links: 4
Link#   Src      Src      Nbr      Nbr
        Index   Interface Index   Interface Link-Cost  Link-Type  Trunk
-----
1       0        Te 48/0/49:1  28      Te 1/0/49:1  500       Pt_Pt Ethernet
2       1        Te 48/0/49:2  30      Te 1/0/49:2  500       Pt_Pt Ethernet
3       2        Te 48/0/49:3  32      Te 1/0/49:3  500       Pt_Pt Ethernet
4       3        Te 48/0/49:4  34      Te 1/0/49:4  500       Pt_Pt Ethernet
```

This example displays link linformation and includes the breakout index of the interface in mixed mode running Network OS v4.0.0:

```
switch# show fabric route linkinfo
Rbridge-id: 1
=====
Reachable:   Yes
Version:     1
No. of Links: 4
Link#   Src      Src      Nbr      Nbr
        Index   Interface Index   Interface Link-Cost  Link-Type  Trunk
-----
1       28       Te 1/0/49  0       Te 48/0/49  500       Pt_Pt Ethernet
2       30       Te 1/0/49  1       Te 48/0/49  500       Pt_Pt Ethernet
3       32       Te 1/0/49  2       Te 48/0/49  500       Pt_Pt Ethernet
4       34       Te 1/0/49  3       Te 48/0/49  500       Pt_Pt Ethernet
Rbridge-id: 48
=====
Reachable:   Yes
Version:     1
No. of Links: 4
Link#   Src      Src      Nbr      Nbr
        Index   Interface Index   Interface Link-Cost  Link-Type  Trunk
-----
1       0        Te 48/0/49  28      Te 1/0/49  500       Pt_Pt Ethernet
2       1        Te 48/0/49  30      Te 1/0/49  500       Pt_Pt Ethernet
3       2        Te 48/0/49  32      Te 1/0/49  500       Pt_Pt Ethernet
4       3        Te 48/0/49  34      Te 1/0/49  500       Pt_Pt Ethernet
```



This example displays link information and includes the breakout index of the interface in mixed mode running Network OS v4.1.0:

```
switch# show fabric route linkinfo
Rbridge-id: 1
=====
Reachable:    Yes
Version:      1
No. of Links: 4
Link#   Src      Src      Nbr      Nbr      Link-Cost  Link-Type  Trunk
      Index  Interface Index  Interface
-----
1       28       Te 1/0/49:1  0       Te 48/0/49  500       Pt_Pt Ethernet
2       30       Te 1/0/49:2  1       Te 48/0/49  500       Pt_Pt Ethernet
3       32       Te 1/0/49:3  2       Te 48/0/49  500       Pt_Pt Ethernet
4       34       Te 1/0/49:4  3       Te 48/0/49  500       Pt_Pt Ethernet
Rbridge-id: 48
=====
Reachable:    Yes
Version:      1
No. of Links: 4
Link#   Src      Src      Nbr      Nbr      Link-Cost  Link-Type  Trunk
      Index  Interface Index  Interface
-----
1       0        Te 48/0/49:1  28      Te 1/0/49  500       Pt_Pt Ethernet
2       1        Te 48/0/49:2  30      Te 1/0/49  500       Pt_Pt Ethernet
3       2        Te 48/0/49:3  32      Te 1/0/49  500       Pt_Pt Ethernet
4       3        Te 48/0/49:4  34      Te 1/0/49  500       Pt_Pt Ethernet
```

## Related Commands

[show fabric route topology](#), [show fabric route neighbor-state](#)

# show fabric route multicast

Displays ISLs that receives any forwarded Broadcast, unknown Unicast, and Multicast (BUM) traffic.

## Syntax

```
show fabric route multicast [ rbridge-id | all ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all switches in the fabric.

## Modes

Privileged EXEC mode

## Usage Guidelines

The multicast routing information indicates all ports that are members of the multicast distribution tree ports that are able to send and receive multicast frames.

If this command is executed without operands, it displays the multicast information of the local RBridge.

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge.

When a fabric is running normal mode (switches are running Network OS 4.1.0 or later), CLI output displays the QSFP breakout index for all switches in the fabric.

When a fabric is running mixed mode (switches are running Network OS 4.1.0 or earlier), the QSFP breakout index is not displayed in the output on switches running Network OS versions earlier than v4.1.0. In such instances, the neighbor port WWN (Nbr-WWN) is displayed as *XX:XX:XX:XX:XX:XX:XX:XX*.

This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.

## Command Output

The **show fabric route multicast** command displays the following information:

Output field	Description
Rbridge-id	RBridge ID of the switch. Valid values range from 1 through 239.
Mcast Priority	Mcast priority value of the switch. Valid values range from 1 through 255.
Enet IP Addr	The switch Ethernet IP address.
WWN	World Wide Name of the switch.
Name	Switch name.
Src-Index	Source index of the local RBridge.

Output field	Description
Src-Port	Source port of the local RBridge in the format "local-rbridge-id/slot/port".
Nbr-Index	Neighbor Index of the ISL connected from local port.
Nbr-Port	Neighbor port of the ISL connected from the local RBridge in the format "nbr-rbridge-id/slot/port".

## Examples

To display the multicast routing information for all ports in the switch:

```
switch# show fabric route multicast
```

```
Root of the Multicast-Tree
```

```
=====
```

```
Rbridge-id: 1
```

```
Mcast Priority: 1
```

```
Enet IP Addr: 10.24.85.212
```

```
WWN: 10:00:00:05:1e:cd:73:fa
```

```
Name: switch
```

```
Rbridge-id: 1
```

Src-Index	Src-Port	Nbr-Index	Nbr-Port	BW	Trunk
7	Te 1/0/7	55	Te 22/0/55	40G	Yes
15	Te 1/0/15	57	Te 23/0/57	60G	Yes
22	Te 1/0/22	58	Te 24/0/58	40G	Yes
26	Te 1/0/26	56	Te 25/0/56	60G	Yes
34	Te 1/0/34	58	Te 26/0/58	60G	Yes
41	Te 1/0/41	59	Te 27/0/59	20G	Yes
44	Te 1/0/44	56	Te 28/0/56	60G	Yes

This example displays route information and includes the breakout index of the interface in normal mode.

```
sw0# show fabric route multicast
```

```
Root of the Multicast-Tree
```

```
=====
```

```
Rbridge-id: 1
```

```
Mcast Priority: 1
```

```
Enet IP Addr: 10.38.19.47
```

```
WWN: 10:00:00:05:33:65:0b:20
```

```
Name: sw0
```

```
Rbridge-id: 48
```

Src-Index	Src-Port	Nbr-Index	Nbr-Port	BW	Trunk
0	Te 48/0/49:1	28	Te 1/0/49:1	0G	

This example displays route information and includes the breakout index of the interface in mixed mode running Network OS v4.0.0.

```
sw0# show fabric route multicast
```

```
Root of the Multicast-Tree
```

```
=====
```

```
Rbridge-id: 1
```

```
Mcast Priority: 1
```

```
Enet IP Addr: 10.38.19.47
```

```
WWN: 10:00:00:05:33:65:0b:20
```

```
Name: sw0
```

```
Rbridge-id: 1
```

Src-Index	Src-Port	Nbr-Index	Nbr-Port	BW	Trunk
28	Te 1/0/49	0	Te 48/0/49	10G	

This example displays route information and includes the breakout index of the interface in mixed mode running Network OS v4.1.0.

```
sw0# show fabric route multicast
Root of the Multicast-Tree
=====
Rbridge-id: 1
Mcast Priority: 1
Enet IP Addr: 10.38.19.47
WWN: 10:00:00:05:33:65:0b:20
Name: sw0
Rbridge-id: 1
Src-Index   Src-Port           Nbr-Index   Nbr-Port       BW    Trunk
-----
0           Te 48/0/49:1      28          Te 1/0/49       10G
```

## Related Commands

[fabric route mcast](#), [show fabric route topology](#), [show running-config fabric route mcast](#), [show fabric route topology](#)

# show fabric route neighbor-state

Displays the state information of all the ISL links connected to the RBridge.

## Syntax

```
show fabric route neighbor-state [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

FSPF defines a neighbor as a remote ISL interface that is directly attached to the local RBridge. If the interfaces are trunked, the command displays data only about the trunk primary.

If no information is available for the switch, the command displays the message "No ISL found."

If no RBridge is specified, the neighbor state information for the local switch is displayed.

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge.

When a fabric is running normal mode (switches are running Network OS 4.1.0 or later), CLI output displays the QSFP breakout index for all switches in the fabric.

When a fabric is running mixed mode (switches are running Network OS 4.1.0 or earlier), the QSFP breakout index is not displayed in the output on switches running Network OS versions earlier than v4.1.0. This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.

## Command Output

The **show fabric route neighbor-state** command displays the following information:

Output field	Description
rbridge-id	ID of the RBridge. Valid values range from 1 through 239.
# ISLs	The number of ISLs that connect the switch to neighbor switches.
Src Index	E_Port interface on the local switch. This value is typically equal to the Index field reported in the switchShow command.

Output field	Description
Src Interface	Physical interface on the local switch in the format "Te rbridge-id/slot/port". If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
Nbr Index	E_Port interface on the remote switch. This value is typically equal to the index field reported in the switchShow command.
Nbr Interface	Physical interface on the remote switch in the format "Te rbridge-id/slot/port". If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
Nbr State	The FSPF neighbor state for the port attached to remote switch. The neighbor can be in one of the following states: <ul style="list-style-type: none"> <li>• NB_ST_DOWN—the neighbor is down.</li> <li>• NB_ST_INIT—the neighbor is initializing.</li> <li>• NB_ST_DB_EX—the neighbor and the switch are exchanging data from their Link State Records (LSR) databases.</li> <li>• NB_ST_DB_ACK_WT—the neighbor is waiting for the switch to acknowledge the LSR database.</li> <li>• NB_ST_DB_WT—the LSR Database is in waiting state; synchronization is in process.</li> <li>• NB_ST_FULL—the neighbor is in the last, finishing state. The E_Port can route frames only if the neighbor is in full state.</li> </ul>

## Examples

To display the state of FSPF neighbors for the local switch:

```
switch# show fabric route neighbor-state

Rbridge-id: 66  #ISLs: 8
  Src      Src      Nbr      Nbr      Nbr
  Index   Interface  Index   Interface  State
-----
  2       Te 66/0/2    2       Te 65/0/2    NB_ST_FULL
  17      Te 66/0/17   41      Te 65/0/41   NB_ST_FULL
  18      Te 66/0/18   42      Te 65/0/42   NB_ST_FULL
  19      Te 66/0/19   43      Te 65/0/43   NB_ST_FULL
  20      Te 66/0/20   44      Te 65/0/44   NB_ST_FULL
  23      Te 66/0/23   47      Te 65/0/47   NB_ST_FULL
  49      Fi 66/0/1    129     Fi 2/-/-     NB_ST_FULL
  53      Fi 66/0/5    129     Fi 160/-/-   NB_ST_FULL
```

This example displays neighbor state route details and includes the breakout index of the interface in normal mode.

```
sw0# show fabric route neighbor-state
Rbridge-id: 1  #ISLs: 1
  Src      Src      Nbr      Nbr      Nbr
  Index   Interface  Index   Interface  State
-----
  30      Te 1/0/49:1  1       Te 48/0/49:1  NB_ST_FULL
```

This example displays neighbor state route details of the non-trunked port in mixed mode running Network OS v4.0.0. (In mixed mode, the QSFP breakout index is not displayed in the output on a switch running Network OS versions earlier than v4.1.0.)

```
sw0# show fabric route neighbor-state
Rbridge-id: 1  #ISLs: 2
  Src      Src      Nbr      Nbr      Nbr
  Index   Interface  Index   Interface  State
-----
  30      Te 1/0/49    1       Te 48/0/49    NB_ST_FULL
```

This example displays neighbor state route details of the non-trunked port in mixed mode running Network OS v4.1.0.

```
sw0# show fabric route neighbor-state
Rbridge-id: 1   #ISLs: 2
Src      Src      Nbr      Nbr      Nbr
Index   Interface   Index   Interface   State
-----
30      Te 1/0/49:1   1      Te 48/0/49   NB_ST_FULL
```

## Related Commands

[show fabric route topology](#), [show fabric route linkinfo](#)

# show fabric route pathinfo

Displays the path of a data stream through a fabric and provides statistics about each hop on that path.

## Syntax

**show fabric route pathinfo VCS ID Domain Source Port Destination Port Basic Stats Extended Stats Reverse Path**

## Parameters

You are prompted to select parameters interactively. The command will prompt you for the following parameters:

### VCS ID

Enter the VCS ID of the destination Network OS switch or the Fabric ID of the destination Fabric OS switch. If unspecified, the value defaults to -1, which specifies the cluster of the local switch.

### Domain

Enter the RBridge ID of the destination Network OS switch or the domain ID of the destination Fabric OS switch. You must enter a value for this parameter. It has no default value.

### Source Port

Enter the port index of the port at the head of the data stream to be traced. If unspecified, the value defaults to -1, which specifies the embedded port.

### Destination Port

Enter the port index of the port on the destination switch for the path being traced. If unspecified, the value defaults to -1, which specifies the embedded port. The command output also reports the status of the Destination Port. If the specified port is out of range on the destination switch, the command fails with the message "Target port not present."

### Basic Stats

Enter **y** to display basic statistics about each hop. By default, basic statistics are not displayed.

### Extended Stats

Enter **y** to display extended statistics about each hop. By default, extended statistics are not displayed.

### Reverse Path

Enter **y** to display reverse path information in addition to the forward path information. By default, reverse path information is not displayed. The path from port A on switch X to port B on switch Y might be different from the path from port B to port A depending on the links traversed between a given sequence of switches, or the reverse path might involve different switches.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge. This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.



Use this command to display detailed routing information and statistics for a data stream from a source port on the local switch to a destination port on another switch. The destination switch can be a member of the same cluster, a member of a different cluster, a member of a Fabric OS backbone fabric, or a member of a Fabric OS edge fabric. This routing information describes the exact path that a user data stream takes to go from the source port to the destination port.

Use this command to check whether a congested link might be causing performance degradation on a specific data stream or path.

You can request statistics for each hop in addition to the routing information. These statistics are presented for the input and output ports for both receive and transmit modes. You can select basic statistics, extended statistics, or both. Statistics are not reported for the embedded port. Some throughput values are reported in multiple time intervals to describe current path utilization and the average throughput over a longer period of time.

To collect these statistics, this command uses a special frame that is sent hop-by-hop from the source switch to the destination switch. To prevent such a frame from looping forever if an error occurs, a maximum of 25 hops is enforced. The hop count includes all hops in the direct path from source to destination, and also all the hops in the reverse path, if the tracing of the reverse path is requested. If the hop limit is exceeded, information collected up to the switch that returned the error is displayed along with the message "Maximum hops exceeded."

## Command Output

Regardless of parameter selection, the **show fabric route pathinfo** command displays the following information of the destination port and routing information about each hop:

Output field	Description
Target port is	Provides the status of the destination port. It can have one of the following values: <ul style="list-style-type: none"> <li>Embedded—this is the embedded port.</li> <li>Not active—the port is not connected or is still initializing and has not yet reached a stable state.</li> <li>E_Port</li> <li>F_Port</li> </ul>
Hop	The hop number. The local switch is hop 0.
In Port	The port index of the port that the frames come in from on this path. For hop 0, this is the source port.
Domain ID	Routing bridge ID of the Network OS switch or domain of the Fabric OS switch.
Out Port	The port index of the port that the frames use to reach the next hop on this path. For the last hop, this is the destination port.
BW	The bandwidth of the output ISL in Gbps. It does not apply to the embedded port.
Cost	The cost of the ISL used by the fabric shortest path first (FSPF) routing protocol. It applies only to an E_Port.

If basic statistics are requested, the following information is provided for each hop in addition to the routing information:

Output field	Description
B/s (1s)	Bytes per second transmitted and received over the previous 1-second period for the in port and for the out port.
B/s (64s)	Bytes per second transmitted and received over the previous 64-second period for the in port and for the out port.
TxCrdz(1s)	The length of time, in milliseconds, over the previous 1 second interval that the port was unable to transmit frames because the transmit BB credit was 0. The purpose of this statistic is to

Output field	Description
	detect congestion or a device affected by latency. This parameter is sampled at 1 millisecond (ms) intervals, and the counter is incremented if the condition is true. Each sample represents 1 ms of time with a 0 Tx BB Credit. An increment of this counter means that the frames could not be sent to the attached device for 1 ms, indicating degraded performance.
TxCrdz(64s)	The length of time, in milliseconds, over the previous 64-second interval that the port was unable to transmit frames because the transmit BB credit was 0.

If extended statistics are requested, the following information is provided for each hop in the data path:

Output field	Description
F/s (1s)	The number of frames received or transmitted per second over the previous 1-second period.
F/s (64s)	The number of frames received or transmitted per second over the previous 64-second period.
Words	The total number of 4-byte Fibre Channel words.
Frames	The total number of frames.
Errors	The total number of errors that may have caused a frame not to be received correctly. This includes cyclic redundancy check (CRC) errors, bad end-of-frame (EOF) errors, frame truncated errors, frame-too-short errors, and encoding errors inside a frame.

## Examples

To show path information without statistics or reverse path information:

```
switch# show fabric route pathinfo

Fabric ID (1..128) [-1]      : 10

Domain      : 1

Source Port [-1]          :
Destination Port [-1]     :
Basic Stats [y/n/yes/no]? : n

Extended Stats [y/n/yes/no]? : n

Reverse Path[y/n/yes/no]?  : n
```

```
-----
Target port is Embedded
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
0        E              152           1             10G     500
1        5              142           54            4G      500
2        14             5             1             4G      10000
3        217            100           793           48G     500
4        1209           2             148           8G      500
5        3              1             E             --      --
```

To show path information with basic statistics:

```
switch# show fabric route pathinfo
```

```
Fabric ID (1..128) [-1]      : 10
```

```
Domain      : 1
```

```
Source Port [-1]      :
```

```
Destination Port [-1]  :
```

```
Basic Stats [y/n/yes/no]? : y
```

```
Extended Stats [y/n/yes/no]? : n
```

```
Reverse Path[y/n/yes/no]? : n
```

```
-----
Target port is Embedded
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  0      E              152            1             10G     500
Port
-----
                        Tx              Rx              Tx              Rx
B/s (1s)      --              --              0              0
B/s (64s)     --              --              0              0
TxCrzd (1s)   --              --              0              --
TxCrzd (64s)  --              --              0              --
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  1      5              142            54            4G      500
Port
                        Tx              Rx              Tx              Rx
B/s (1s)      0              0              0              0
B/s (64s)     0              0              7              7
TxCrzd (1s)   0              --              0              --
TxCrzd (64s)  0              --              0              --
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  2      14             5              1             4G     10000
Port
                        Tx              Rx              Tx              Rx
B/s (0s)      0              0              0              0
B/s (0s)      0              0              0              0
TxCrzd (0s)   0              --              0              --
TxCrzd (0s)   0              --              0              --
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  3      217            100            793            48G     500
Port
                        Tx              Rx              Tx              Rx
B/s (1s)      0              0              0              0
B/s (64s)     4              4              0              0
TxCrzd (1s)   0              --              0              --
TxCrzd (64s)  0              --              0              --
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  4      1209           2              148            8G      500
Port
                        Tx              Rx              Tx              Rx
B/s (1s)      0              0              0              0
B/s (64s)     0              0              3              0
TxCrzd (1s)   0              --              0              --
TxCrzd (64s)  0              --              0              --
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  5      3              1              E              --      --
-----
```

show fabric route pathinfo

Port	3		E	
	Tx	Rx	Tx	Rx
B/s (1s)	0	0	--	--
B/s (64s)	0	3	--	--
TxCrdz (1s)	0	--	--	--
TxCrdz (64s)	0	--	--	--

To show path information with extended statistics and reverse path information:

```
switch# show fabric route pathinfo
```

```
Fabric ID (1..128) [-1]      : 10
```

```
Domain      : 1
```

```
Source Port [-1]      :
```

```
Destination Port [-1]  :
```

```
Basic Stats [y/n/yes/no]? : y
```

```
Extended Stats [y/n/yes/no]? : y
```

```
Reverse Path[y/n/yes/no]? : y
```

```
-----
Target port is Embedded
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  0      E              152            1             10G     500
Port
          Tx          Rx              Tx          Rx
-----
B/s (1s)      --          --              0             0
B/s (64s)     --          --              0             0
TxCrzd (1s)   --          --              0             --
TxCrzd (64s)  --          --              0             --
F/s (1s)      --          --              0             0
F/s (64s)     --          --              0             0
Words         --          --              0             0
Frames        --          --              0             0
Errors        --          --              --            0
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  1      5              142            54            4G     500
Port
          Tx          Rx              Tx          Rx
-----
B/s (1s)      0             0              0             0
B/s (64s)     0             0              7             7
TxCrzd (1s)   0             --              0             --
TxCrzd (64s)  0             --              0             --
F/s (1s)      0             0              0             0
F/s (64s)     0             0              0             0
words         0             0              967           967
Frames        0             0              1204          967
Errors        --            0              --            0
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  2      14             5              1             4G    10000
Port
          Tx          Rx              Tx          Rx
-----
B/s (0s)      0             0              0             0
B/s (0s)      0             0              0             0
TxCrzd (0s)   0             --              0             --
TxCrzd (0s)   0             --              0             --
F/s (0s)      0             0              0             0
F/s (0s)      0             0              0             0
words         0             0              0             0
Frames        0             0              0             0
Errors        --            0              --            0
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  3      217            100            793           48G     500
Port
          Tx          Rx              Tx          Rx
-----
B/s (1s)      0             0              0             0
B/s (64s)     4             4              0             0
-----
```

show fabric route pathinfo

```

TxCrzd (1s)          0          --          0          --
TxCrzd (64s)         0          --          0          --
F/s (1s)             0          0          0          0
F/s (64s)            0          0          0          0
words                50570        50570        511118479    511118479
Frames               50742        50570        539255694    511118479
Errors               --          0          --          0
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  4      1209          2          148          8G      500
Port
          Tx          1209      Rx          Tx          148      Rx
-----
B/s (1s)          0          0          0          0
B/s (64s)         0          0          3          0
TxCrzd (1s)       0          --          0          --
TxCrzd (64s)      0          --          0          --
F/s (1s)          0          0          0          0
F/s (64s)         0          0          0          0
words             608          608          424          424
Frames            454          608          563          424
Errors            --          0          --          0
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  5      3          1          E          --      --
Port
          Tx          3          Rx          Tx          E          Rx
-----
B/s (1s)          0          0          --          --
B/s (64s)         0          3          --          --
TxCrzd (1s)       0          --          --          --
TxCrzd (64s)      0          --          --          --
F/s (1s)          0          0          --          --
F/s (64s)         0          0          --          --
words            1244          1244          --          --
Frames            898          1244          --          --
Errors            --          0          --          --
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  6      E          1          0          8G      500
Port
          Tx          E          Rx          Tx          0          Rx
-----
B/s (1s)          --          --          0          0
B/s (64s)         --          --          4          4
TxCrzd (1s)       --          --          0          --
TxCrzd (64s)      --          --          0          --
F/s (1s)          --          --          0          0
F/s (64s)         --          --          0          0
Words             --          --          809          809
Frames            --          --          1645          809
Errors            --          --          --          0
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  7      149          2          1204          48G      500
Port
          Tx          149      Rx          Tx          1204      Rx
-----
B/s (1s)          0          0          0          0
B/s (64s)         4          4          0          0
TxCrzd (1s)       0          --          0          --
TxCrzd (64s)      0          --          0          --
F/s (1s)          0          0          0          0
F/s (64s)         0          0          0          0
words             707          707          56          56
Frames            403          707          57          56
Errors            --          0          --          0
Hop      In Port      Domain ID      Out Port      BW      Cost
-----
  8      796          100          217          4G      500
Port
          Tx          796      Rx          Tx          217      Rx
-----

```

	Tx	Rx	Tx	Rx
B/s (1s)	0	0	0	0
B/s (64s)	0	0	4	4
TxCrdz (1s)	0	--	0	--
TxCrdz (64s)	0	--	0	--
F/s (1s)	0	0	0	0
F/s (64s)	0	0	0	0
words	48267544	48267544	50570	50570
Frames	1164982	48267544	50742	50570
Errors	--	0	--	0

## Related Commands

[show fabric route topology](#), [show fabric route neighbor-state](#)

# show fabric route topology

Displays the RBridge routes from the source switch to destination switches.

## Syntax

```
show fabric route topology [ src-rbridged src-id ] [ dst-rbridged dst_id ]
```

## Parameters

### src-rbridged

Specifies the source RBridge ID

*src-id*

Specifies details on the source RBridge.

### dst-rbridged

Specifies the destination RBridge ID

*dst\_id*

Specifies details on the destination RBridge.

## Modes

Privileged EXEC mode

## Usage Guidelines

The RBridge routes to other switches are the available paths to remote domains that unicast traffic can take.

The source RBridge ID must be the local RBridge ID in this release. It is an optional parameter. If you do not specify the source RBridge ID or the destination RBridge ID, the system routes to all destinations in the Fabric.

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge.

When a fabric is running normal mode (switches are running Network OS 4.1.0 or later), CLI output displays the QSFP breakout index for all switches in the fabric.

When a fabric is running mixed mode (switches are running Network OS 4.1.0 or earlier), the QSFP breakout index is not displayed in the output on switches running Network OS versions earlier than v4.1. This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.

## Command Output

The **show fabric route topology** command displays the following information:

Output field	Description
Path Count	The number of currently active paths to the destination domain.
Src RB-ID	RBridge ID of the source switch. Valid values range from 1 through 239.
Dst RB-ID	Destination rbridge-id to where the traffic flows. Valid values range from 1 through 255.



Output field	Description
Out Index	The port index to which incoming frames are forwarded to reach the destination RBridge.
Out Interface	The port interface (local-rbridge-id/slot/port) of the local RBridge to which incoming frames are forwarded to the destination RBridge. If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
ECMP Grp	The Equal Cost MultiPath group
Hops	The maximum number of hops to reach destination RBridge.
Cost	The cost of reaching destination domain.
Nbr Index	Neighbor Index of the ISL connected from local port.
Nbr Interface	Neighbor interface of the ISL connected from the local RBridge in the format "nbr-rbridge-id/slot/port". If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
BW	Bandwidth of the traffic.
Trunk	Displays "Yes" if trunk is enabled in the ISL.

## Examples

To display the fabric route topology information:

```
switch# show fabric route topology
Total Path Count: 4
```

Src RB-ID	Dst RB-ID	Out Index	Out Interface	ECMP Grp	Hops	Cost	Nbr Index	Nbr Interface	BW	Trunk
1	2	17	Te 1/8/18	1	2	1000	25	Te 22/0/18	20G	Yes
	3	17	Te 1/8/18	1	3	11000	25	Te 22/0/18	20G	Yes
	22	17	Te 1/8/18	1	1	500	25	Te 22/0/18	20G	Yes
	160	17	Te 1/8/18	1	2	1000	25	Te 22/0/18	20G	Yes

This example displays route topology details and includes the breakout index of the neighbor interface in normal mode.

```
switch# show fabric route topology
Total Path Count: 3
```

Src RB-ID	Dst RB-ID	Out Index	Out Interface	ECMP Grp	Hops	Cost	Nbr Index	Nbr Interface	BW	Trunk
1	48	30	Te 1/0/49:1	1	1	500	1	Te 48/0/49:1	40G	Yes

This example displays route topology details and includes the breakout index of the neighbor interface in mixed mode running Network O.S. v4.0.0.

```
switch# show fabric route topology
Total Path Count: 3
```

Src RB-ID	Dst RB-ID	Out Index	Out Interface	ECMP Grp	Hops	Cost	Nbr Index	Nbr Interface	BW	Trunk
1	48	30	Te 1/0/49	1	1	500	1	Te 48/0/49	40G	Yes

This example displays route topology details and includes the breakout index of the neighbor interface in mixed mode running Network O.S. v4.1.0.

```
switch# show fabric route topology
Total Path Count: 3
```

Src RB-ID	Dst RB-ID	Out Index	Out Interface	ECMP Grp	Hops	Cost	Nbr Index	Nbr Interface	BW	Trunk
1	48	30	Te 1/0/49:1	1	1	500	1	Te 48/0/49	40G	Yes

show fabric route topology

## Related Commands

[fabric route mcast](#)

# show fabric trunk

Displays trunk information of Inter-Switch Link (ISL) ports.

## Syntax

```
show fabric trunk [ rbridge-id | all ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all switches in the fabric.

## Modes

Privileged EXEC mode

## Usage Guidelines

If this command is executed without operands, it displays the trunk information of the local RBridge.

The **show fabric** family of commands display neighbor and local port information when connected to a down-level RBridge.

When a fabric is running normal mode (switches are running Network OS 4.1.0 or later), CLI output displays the QSFP breakout index for all switches in the fabric.

When a fabric is running mixed mode (switches are running Network OS 4.1.0 or earlier), the QSFP breakout index is not displayed in the output on switches running Network OS versions earlier than v4.1.0. In such instances, the neighbor port WWN (Nbr-WWN) is displayed as *XX:XX:XX:XX:XX:XX:XX:XX*.

This command displays ISL details, including the breakout index of the interface if breakout mode is configured on the source or neighbor interface.

## Command Output

The **show fabric trunk** command displays the following information:

Output field	Description
Rbridge-id	RBridge ID of the switch. Valid values range from 1 through 239.
Trunk Group	Displays each trunking group on a switch. All ports that are part of this trunking group are displayed.
Src Index	Source index of the local RBridge.
Src Interface	Source interface of the local RBridge in the format "local-rbridge-id/slot/port". If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
Nbr Index	Neighbor Index of the ISL connected from local port. If the link is segmented and the neighbor RBridge details are unavailable, "?" displays in this field.

Output field	Description
Nbr Interface	Neighbor interface of the ISL connected from the local RBridge in the format "nbr-rbridge-id/slot/port". If the ISL is not completely up, this field will be displayed as "?/?/?". If the ISL is not up, then "?/?/?" displays for a Te interface and "-/-/" displays for an Fi interface.
Nbr-WWN	Neighbor WWN of the switch. If the ISL is segmented and the neighbor RBridge details are not available, then "?:?:?:?:?:?:?:?:?:?:?" displays in this field.

## Examples

To display the fabric trunk information:

```
switch# show fabric trunk
```

Possible completions:

```
all          all RBridges in fabric
rbridge-id  Syntax: rbridge-id [rbridge-id]
|           Output modifiers
<cr>
switch# show fabric trunk
```

```
Rbridge-id: 65
Trunk Src   Source      Nbr   Nbr
Group Index Interface  Index Interface  Nbr-WWN
-----
1      1      Te 65/0/1      1     Te 66/0/1      10:00:00:05:33:6F:27:57
1      2      Te 65/0/2      2     Te 66/0/2      10:00:00:05:33:6F:27:57
2      45     Te 65/0/45     21    Te 66/0/21     10:00:00:05:33:6F:27:57
2      47     Te 65/0/47     23    Te 66/0/23     10:00:00:05:33:6F:27:57
2      46     Te 65/0/46     22    Te 66/0/22     10:00:00:05:33:6F:27:57
```

This example displays trunk details and includes the breakout index of the neighbor interface in normal mode.

```
sw0# show fabric trunk
Rbridge-id: 1
Trunk Src   Source      Nbr   Nbr
Group Index Interface  Index Interface  Nbr-WWN
-----
1      74      Te 1/0/49:1    0     Te 48/0/49:1    XX:XX:XX:XX:XX:XX:XX:XX
1      76      Te 1/0/49:2    1     Te 48/0/49:2    XX:XX:XX:XX:XX:XX:XX:XX
1      80      Te 1/0/49:3    3     Te 48/0/49:3    XX:XX:XX:XX:XX:XX:XX:XX
1      78      Te 1/0/49:4    2     Te 48/0/49:4    XX:XX:XX:XX:XX:XX:XX:XX
Where XX:XX:XX:XX:XX:XX:XX:XX is the neighbor port WWN
```

This example displays trunk details and includes the breakout index of the neighbor interface in mixed mode running Network OS v4.0.0.

```
sw0# show fabric trunk
Rbridge-id: 1
Trunk Src   Source      Nbr   Nbr
Group Index Interface  Index Interface  Nbr-WWN
-----
1      74      Te 1/0/49      0     Te 48/0/49      XX:XX:XX:XX:XX:XX:XX:XX
1      76      Te 1/0/49      1     Te 48/0/49      XX:XX:XX:XX:XX:XX:XX:XX
1      80      Te 1/0/49      3     Te 48/0/49      XX:XX:XX:XX:XX:XX:XX:XX
1      78      Te 1/0/49      2     Te 48/0/49      XX:XX:XX:XX:XX:XX:XX:XX
```

This example displays trunk details and includes the breakout index of the neighbor interface in mixed mode running Network OS v4.1.0.

```
sw0# show fabric trunk
Rbridge-id: 1
Trunk Src      Source      Nbr      Nbr
Group Index     Interface   Index     Interface Nbr-WWN
-----
1      74      Te 1/0/49:1 0      Te 48/0/49 XX:XX:XX:XX:XX:XX:XX:XX
1      76      Te 1/0/49:2 1      Te 48/0/49 XX:XX:XX:XX:XX:XX:XX:XX
1      80      Te 1/0/49:3 3      Te 48/0/49 XX:XX:XX:XX:XX:XX:XX:XX
1      78      Te 1/0/49:4 2      Te 48/0/49 XX:XX:XX:XX:XX:XX:XX:XX
```

## Related Commands

[fabric trunk enable](#), [show fabric isl](#), [show fabric islports](#)

# show fcoe devices

Displays the Ethernet port, WWN, and MAC address of participating FCoE devices.

## Syntax

**show fcoe devices**

## Modes

Privileged EXEC mode

## Command Output

The **show fcoe devices** command displays the following information:

Output field	Description
Eth-port	Ethernet port
Device MAC	Device MAC address

## Examples

To display parameters for FCoE devices:

```
device# show fcoe devices
```

```
=====
Eth-port      Device MAC
=====
Te 1/0/1      00:05:1e:8f:f9:de
Te 21/0/1     00:05:1e:8f:f9:df
Te 5/0/1      00:05:1e:8f:f7:df
Po 457        00:05:1e:8f:f8:df
```

## History

Release version	Command history
Network OSv6.0.0	This command was introduced.

# show fcoe fabric-map

Displays the FCoE fabric-map configuration globally in a fabric, or on a single RBridge.

## Syntax

```
show fcoe fabric-map [ rbridge-id rbridge-id | map_name ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

*map-name*

Specifies a user-defined fabric map.

## Modes

Privileged EXEC mode

## Examples

To display the fabric map configuration on all RBridges in the fabric:

```
device# show fcoe fabric-map
```

```
=====
Fabric-Map          VLAN      VFID      Pri    FCMAP      FKA      Timeout
=====
default             1002     128 [D]   3 [D]  0xefc00 [D] 8000 [D]  Enabled [D]
SanA                2008     128 [D]   3 [D]  0xefc10      8000 [D]  Enabled [D]
SanB                2009     128 [D]   3 [D]  0xefc20      8000 [D]  Enabled [D]
=====
```

Total number of Fabric Maps = 3

To display the fabric map configuration for a single RBridge:

```
device# show fcoe fabric-map rbridge-id 19
```

```
=====
Fabric-Map          VLAN      VFID      Pri    FCMAP      FKA      Timeout
=====
default             1002 [D]  128 [D]   3 [D]  0xefc00 [D] 8000 [D]  Enabled [D]
=====
```

To display the fabric map configuration for a specific fabric map:

```
device# show fcoe fabric-map SanA
```

```
=====
Fabric-Map          VLAN      VFID      Pri    FCMAP      FKA      Timeout
=====
SanA                2008     128 [D]   3 [D]  0xefc10      8000 [D]  Enabled [D]
=====
```

## Command Output

The **show fcoe fabric-map** command displays the following information:

Output field	Description
Fabric-Map	Name of the fabric map
VLAN	VLAN ID
VFID	FCoE virtual fabric ID
Pri	Priority
FCMAP	Value of the FCoE MAC address prefix
FKA	FIP keep-alive message interval (D = default)
Timeout	State of FKA timeout messaging (D = default)

## History

Release version	Command history
6.0.0	This command was introduced.

## Related Commands

[fcoe-enodes](#), [show fcoe fcoe-enodes](#)



# show fcoe fcf-group

Displays information about the FCoE Forwarder (FCF) group.

## Syntax

```
show fcoe fcf-group [ rbridge-id rbridge-id | group-name ]
```

## Parameters

**rbridge-id***rbridge-id*

Specifies an RBridge.

*group-name*

Specifies an FCF group.

## Modes

Privileged EXEC mode

## Command Output

The **show fcoe fcf-group** command displays the following information:

Output field	Description
FCF-Group	Name of the FCF group
Fabric-Map	Name of the FCoE map
FCF_RBID	RBridge ID of the FCF device (AG)
FIF_RBID	RBridge ID of the FIF device

## Examples

To display information about FCF groups:

```
device# show fcoe fcf-group
```

```
=====
FCF-Group   Fabric-Map  FCF_RBID      FIF_RBID (s)
=====
SanA        fabmap1     1
rack1       SanA        0

Total number of FCF Groups = 2
```

## History

Release version	Command history
6.0.0	This command was introduced.

# show fcoe fcoe-enodes

Displays the FCoE enodes information for an RBridge ID or all RBridge IDs in the cluster.

## Syntax

```
show fcoe fcoe-enodes [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*  
Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

The **show fcoe fcoe-enodes** command displays the following information:

```
switch# show fcoe fcoe-enodes

=====
Rbridge-id          Fcoe-enodes
=====
1                   64 [D]
2                   70
20                  100
26                  250
37                  1000
100                 0
Total number of fcoe-enodes in the cluster = 1484
```

The **show fcoe fcoe-enodes rbridge-id** command displays the following information:

```
switch# show fcoe fcoe-enodes rbridge-id 20

=====
Rbridge-id          Fcoe-enodes
=====
20                  100
Total number of fcoe-enodes in the cluster = 100.
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.0	This description was modified.

## Related Commands

[show fcoe fabric-map, fcoe-enodes](#)

# show fcoe fcoe-map

Displays information about the FCoE map.

## Syntax

```
show fcoe fcoe-map [ rbridge-id rbridge-id | map_name ]
```

## Parameters

**rbridge-id***rbridge-id*

Specifies an RBridge.

*map\_name*

Specifies an FCoE map.

## Modes

Privileged EXEC mode

## Command Output

The **show fcoe fcoe-map** command displays the following information:

Output field	Description
FCoE-Map	Name of the FCoE map
CEE-Map	Name of the CEE map (always "default")
Fabric-Map	Name of the fabric map

## Examples

To display information about FCoE maps:

```
device# show fcoe fcoe-map
```

```
=====
FCoE-Map          CEE-Map          Fabric-Map
=====
default           default          default
SanA              default          SanA
Total number of Maps = 2
```

## History

Release version	Command history
6.0.0	This command was introduced.

# show fcoe fcport-group

Displays the RBridge IDs and FCoE port connector groups associated with a fabric map.

## Syntax

`show fcoe fcport-group [ rbridge-id rbridge-id ]`

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

This example displays a fabric map with associated RBridge IDs and connector groups.

```
device# show fcoe fcport-group
=====
Fabric-Map          Rbridge-Id          Connector Group
=====
SanA                26                  26/0/1    26/0/3,
                   27                  26/0/5
                   27                  27/0/5
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show fcoe interface

Displays a synopsis of the FCoE interfaces.

## Syntax

```
show fcoe interface [ brief | ethernet [ rbridge-id rbridge-id ] ]
```

## Parameters

### brief

Displays a brief synopsis of the configuration status.

### ethernet

Displays the FCoE Ethernet interface information.

### rbridge-id *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To display brief information about FCoE interfaces:

```
device# show fcoe interface brief
```

```
=====
FCOE IF      Mode      Status      Ethernet-Port      Enode-Binding      #VN
           Cfg  Cur  Config      Prot Typ Interface      Typ Mac-address      Ports
=====
1/6/1      VF   VF  Up          Up   Dyn Rb 5          Dyn 00:05:1e:8f:f9:e7 1
Total number of Interfaces : 1
```

To display information about FCoE Ethernet interfaces:

```
device# show fcoe interface ethernet
```

```
=====
Interface      Fcoe-Map      Status
=====
Te 5/0/33      SanA          Active
Te 6/0/1       SanA          Inactive
Total number of FCOE Ethernet Interfaces : 2
```

## History

Release version	Command history
6.0.0	The command output was modified.

# show fcoe login

Displays FCoE login information.

## Syntax

```
show fcoe login [ interface <N>gigabitethernet rbridge-id/slot/port | rbridge-id rbridge-id ] [ vfid vf_id ] [ vlan vlan_id ]
```

## Parameters

### interface

Specifies an interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### vfid vf\_id

Displays logins for an FCoE virtual fabric. Range is from 1 through 4096.

### vlan vlan\_id

Displays logins for an FCoE VLAN.

## Modes

Privileged EXEC mode

## Usage Guidelines

### NOTE

The FCoE virtual fabric is not to be confused with the Virtual Fabrics feature that supports service or transport VFs.

## Command Output

The **show fcoe login** command displays the following information:

Output field	Description
FCOE-Port	FCoE port number
Eth-port/rbid	Ethernet port number or RBridge ID
Device WWN	Device WWN

Output field	Description
Device MAC	Device MAC address
Session MAC	Session MAC address

## Examples

To display information about all FCoE logins:

```
device# show fcoe login
```

```
=====
FCOE-Port   Eth-port/rbid  Device WWN           Device MAC           Session MAC
=====
Fcoe 1/1/1   Te 1/0/1       10:00:00:05:1e:8f:f9:de  00:05:1e:8f:f9:de  0e:fc:00:01:02:c0
Fcoe 1/21/1  Te 21/0/1      10:00:00:05:1e:8f:f9:df  00:05:1e:8f:f9:df  0e:fc:00:15:05:00
Fcoe 1/21/2  Rb 5           10:00:00:05:1e:8f:f7:df  00:05:1e:8f:f7:df  0e:fc:10:15:06:00
Fcoe 1/21/3  Rb 5           10:00:00:05:1e:84:f8:df  00:05:1e:8f:f8:df  0e:fc:10:15:07:00
=====
```

To display information about a login on a TenGigabitEthernet port:

```
device# show fcoe login interface tengigabitethernet 1/0/1
```

```
=====
FCOE-Port   Eth-port   Device WWN           Device MAC           Session MAC
=====
Fcoe 1/1/1   Te 1/0/1    10:00:00:05:1e:8f:f9:de  00:05:1e:8f:f9:de  0e:fc:00:01:02:c0
=====
```

To display information about a login on a Port-channel port:

```
device# show fcoe login interface port-channel 179
```

```
=====
FCOE-Port   Eth-port   Device WWN           Device MAC           Session MAC
=====
Fcoe 1/1/1   Po 179     10:00:00:05:1e:8f:f9:de  00:05:1e:8f:f9:de  0e:fc:00:01:02:c0
=====
```

## History

Release version	Command history
6.0.0	This command description was modified.

# show fcsp auth-secret dh-chap

Displays the device's WWN for which a shared secret is configured.

## Syntax

```
show fcsp auth-secret dh-chap
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported on the Extreme 6740, VDX 2741, and VDX 2746.

## Examples

```
device# show fcsp auth-secret dh-chap
10:00:00:05:1e:7a:c3:00
```

## Related Commands

[fcsp auth](#), [fcsp auth-secret dhchap](#), [show running-config fcsp auth](#)



# show fibrechannel login

Displays Fibre Channel logins or a specified fibrechannel interface or a specified RBridge ID..

## Syntax

```
show fibrechannel login [ rbridge-id rbridge-id | interface fcoe-id ]
```

## Parameters

*rbridge-id rbridge-id*

Specifies an RBridge ID.

*interface fcoe-id*

Specifies the FCoE interface name in VN-Num/RBridge-ID/port format of the Fibre Channbel logins to display.

## Modes

Privileged EXEC mode

## Examples

To view all fibrechannels:

```
sw0# show fibrechannel login
```

```
=====
Interface   Index  PID      status      protocol  speed      PortWWN
=====
Fi 17/0/3   2      110146   up(In_Sync ) up        4G Auto    2e:83:00:05:33:26:14:26
Fi 17/0/3   2      110145   up(In_Sync ) up        4G Auto    2d:07:00:05:33:26:14:26
Fi 17/0/3   2      110144   up(In_Sync ) up        4G Auto    2e:fc:00:05:33:26:14:26
Fi 17/0/3   2      110143   up(In_Sync ) up        4G Auto    20:74:00:05:33:26:14:26
Fi 17/0/3   2      110142   up(In_Sync ) up        4G Auto    26:0a:00:05:33:26:14:26
Fi 17/0/3   2      110141   up(In_Sync ) up        4G Auto    28:2d:00:05:33:26:14:26
Fi 17/0/3   2      110140   up(In_Sync ) up        4G Auto    10:00:00:05:33:26:14:26
Fi 17/0/4   3      110100   up(In_Sync ) up        4G Auto    10:00:00:05:33:26:14:27
Total number of Login(s) = 8
```

To view fibrechannel logins for an RBridge:

```
sw0# show fibrechannel login rbridge-id 17
```

```
=====
Interface   Index  PID      status      protocol  speed      PortWWN
=====
Fi 17/0/3   2      110146   up(In_Sync ) up        4G Auto    2e:83:00:05:33:26:14:26
Fi 17/0/3   2      110145   up(In_Sync ) up        4G Auto    2d:07:00:05:33:26:14:26
Fi 17/0/3   2      110144   up(In_Sync ) up        4G Auto    2e:fc:00:05:33:26:14:26
Fi 17/0/3   2      110143   up(In_Sync ) up        4G Auto    20:74:00:05:33:26:14:26
Fi 17/0/3   2      110142   up(In_Sync ) up        4G Auto    26:0a:00:05:33:26:14:26
Fi 17/0/3   2      110141   up(In_Sync ) up        4G Auto    28:2d:00:05:33:26:14:26
Fi 17/0/3   2      110140   up(In_Sync ) up        4G Auto    10:00:00:05:33:26:14:26
Fi 17/0/4   3      110100   up(In_Sync ) up        4G Auto    10:00:00:05:33:26:14:27
```

Total number of Login(s) = 8

show fibrechannel login

To view fibrechannel logins on an interface:

```
sw0# show fibrechannel login interface 17/0/3
```

```
=====
Interface   Index  PID      status      protocol  speed      PortWWN
=====
Fi 17/0/3   2      110146   up(In_Sync ) up        4G Auto    2e:83:00:05:33:26:14:26
Fi 17/0/3   2      110145   up(In_Sync ) up        4G Auto    2d:07:00:05:33:26:14:26
Fi 17/0/3   2      110144   up(In_Sync ) up        4G Auto    2e:fc:00:05:33:26:14:26
Fi 17/0/3   2      110143   up(In_Sync ) up        4G Auto    20:74:00:05:33:26:14:26
Fi 17/0/3   2      110142   up(In_Sync ) up        4G Auto    26:0a:00:05:33:26:14:26
Fi 17/0/3   2      110141   up(In_Sync ) up        4G Auto    28:2d:00:05:33:26:14:26
Fi 17/0/3   2      110140   up(In_Sync ) up        4G Auto    10:00:00:05:33:26:14:26
Total number of Login(s) = 7
```

# show file

Displays the contents of a file in the local flash memory.

## Syntax

```
show file filename
```

## Parameters

*filename*

The name of the file to be displayed.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local device.

## Examples

To display the contents of the startup configuration file:

```
switch# show file startup-config-copy

diag post rbridge-id 237
no enable
!
linecard 2 LC48x10G
linecard 4 LC48x10G
class-map default
match any
!
logging rbridge-id 237
  raslog console INFO
!
logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
logging syslog-facility local LOG_LOCAL7
switch-attributes 237
chassis-name VDX8770-4
host-name sw0
!
support rbridge-id 237
ffdc
!
snmp-server contact "Field Support."
snmp-server location "End User Premise."
snmp-server sys-descr "VDX Switch."
snmp-server community ConvergedNetwork
snmp-server community OrigEquipMfr rw
snmp-server community "Secret C0de" rw
snmp-server community common
snmp-server community private rw
snmp-server community public
snmp-server host 172.26.3.84 public
udp-port 5000
severity-level Info
!
(Output truncated)
```

# show fips

Displays the current FIPS configuration.

## Syntax

```
show fips
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display whether FIPS self tests are enabled, and whether the root account is disabled.

## Examples

To display the FIPS enabled status:

```
switch# show fips

FIPS Selftests: Enabled
Root account:   Disabled
```

# show firmwaredownloadhistory

Displays the firmware download history for the switches.

## Syntax

```
show firmwaredownloadhistory [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

The log records the date and time of the firmware download, the switch name, slot number, process ID and firmware version.

Use this command to display information for the local management module only.

## Examples

To display the firmware download history:

```
switch# show firmwaredownloadhistory
Firmware version history
Sno  Date & Time                Switch Name  Slot  PID   OS Version
1    Thu May  2 05:00:08 2013      sw0         0     1561  nos4.0.0
2    Wed May  1 07:44:43 2013      sw0         0     1551  nos3.0.1
```

## Related Commands

[show version](#)

# show firmwaredownloadstatus

Displays the firmware download activity log.

## Syntax

```
show firmwaredownloadstatus [ brief ] [ summary ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### brief

Displays only the last entry of the firmware download event log.

### summary

Displays a high-level summary of the firmware download status.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display an event log that records the progress and status of events that occur during a firmware download. The event log is created by the **firmware download** command and is retained until you issue another **firmware download** command. A time stamp is associated with each event.

The output varies depending on the hardware platform.

## Examples

To display the firmware download event log on an Extreme VDX 8770-4:

```
device# show firmwaredownloadstatus

[1]: Tue Mar  6 04:05:20 2012
Slot M1: Firmware install begins.

[2]: Tue Mar  6 04:09:02 2012
Slot M1: Firmware install ends.

[3]: Tue Mar  6 04:09:02 2012
Slot M2: Firmware install begins.

[4]: Tue Mar  6 04:12:08 2012
Slot M2: Firmware install ends.

[5]: Tue Mar  6 04:12:09 2012
Slot M1: Firmware starts to swap.

[6]: Tue Mar  6 04:12:09 2012
Slot M2: Firmware starts to swap.
(Output truncated)
```

To display a condensed version of the firmware download status:

```
device# show firmwaredownloadstatus brief

[35]: Tue Mar  6 04:23:10 2012
Slot M1: Firmware is downloaded successfully.
```

To display a high-level summary of the firmware download status:

```
device# show firmwaredownloadstatus summary rbridge-id 1-4
Rid 1: INSTALLING
Rid 2: INSTALLED (Ready for activation)
Rid 3: COMMITTING
Rid 4: COMMITED
```



# show global-running-config

Displays the global running configuration for a node.

## Syntax

```
show global-running-config
```

## Modes

Privileged EXEC mode

## Examples

The following example shows partial output for this command:

```
switch# show global-running-config
logging raslog console INFO
logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
logging syslog-facility local LOG_LOCAL7
no support autoupload enable
support ffdc
snmp-server contact "Field Support."
snmp-server location "End User Premise."
snmp-server sys-descr "VDX Switch."
snmp-server community ConvergedNetwork
snmp-server community OrigEquipMfr rw
snmp-server community "Secret C0de" rw
snmp-server community common
snmp-server community private rw
snmp-server community public
snmp-server user snmpadmin1 groupname snmpadmin
snmp-server user snmpadmin2 groupname snmpadmin
snmp-server user snmpadmin3 groupname snmpadmin
snmp-server user snmpuser1
snmp-server user snmpuser2
snmp-server user snmpuser3
line vty
  exec-timeout 10
!
zoning enabled-configuration cfg-name ""
zoning enabled-configuration default-zone-access allaccess
zoning enabled-configuration cfg-action cfg-save
role name admin desc Administrator
role name user desc User
aaa authentication login local
aaa accounting exec default start-stop none
aaa accounting commands default start-stop none
service password-encryption
username admin password "BwrsDbB+tABWGwPINOVKoQ==\n" encryption-level 7 role admin desc Administrator
username user password "BwrsDbB+tABWGwPINOVKoQ==\n" encryption-level 7 role user desc User
cee-map default
  precedence 1
  priority-group-table 1 weight 40 pfc on
  priority-group-table 15.0 pfc off
  priority-group-table 15.1 pfc off
  priority-group-table 15.2 pfc off
  priority-group-table 15.3 pfc off
  priority-group-table 15.4 pfc off
  priority-group-table 15.5 pfc off
  priority-group-table 15.6 pfc off
  priority-group-table 15.7 pfc off
  priority-group-table 2 weight 60 pfc off
  priority-table 2 2 2 1 2 2 2 15.0
  remap fabric-priority priority 0
  remap lossless-priority priority 0
!
fcoe
  fabric-map default
    vlan 1002
    priority 3
    virtual-fabric 128
    fcmap 0E:FC:00
    max-enodes 64
    advertisement interval 8000
    keep-alive timeout
  !
  map default
    fabric-map default
    cee-map default
  !
```

```
!  
interface Vlan 1  
  shutdown  
!  
interface Vlan 123  
  shutdown  
  protocol lldp  
  advertise dcbx-fcoe-app-tlv  
  advertise dcbx-fcoe-logical-link-tlv  
  advertise dcbx-tlv  
  system-description VDX-VCS 300  
!  
vlan dot1q tag native  
port-profile default  
vlan-profile  
  switchport  
  switchport mode trunk  
  switchport trunk allowed vlan all  
  switchport trunk native-vlan 1  
!  
class-map cee  
class-map default
```

## Related Commands

[show rbridge-running config](#), [show rbridge-local-running-config](#)

# show ha

Displays the High Availability (HA) status of the management modules.

## Syntax

```
show ha [ all-partitions ] [ rbridge-id { rbridge-id } all ] ]
```

## Parameters

### all-partitions

Displays the HA status for all partitions.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

If a failover becomes necessary while the management modules are not in sync, the standby management module reboots, and the failover is disruptive.

## Command Output

The **show ha** command displays the following information:

Output field	Description
Local MM state	warm or cold, recovering or recovered
Remote MM state	warm or cold, recovering or recovered, or not available
High Availability	enabled or disabled
Heartbeat	up or down
Health of standby management module	<p>The health of the standby CP is defined as follows:</p> <ul style="list-style-type: none"> <li>• <b>Healthy</b>—The standby management module is running and the background health diagnostic has not detected any errors.</li> <li>• <b>Failed</b>—The standby is running, but the background health diagnostic has discovered a problem with the blade. Check the logs to determine the appropriate action.</li> <li>• <b>Unknown</b>—The standby management module's health state is unknown because of one of the following reasons: the standby CP does not exist, Heartbeat is down, or the Health Monitor has detected a configuration file error.</li> </ul>

Output field	Description
HA synchronization status	<p>The High Availability synchronization status is defined as follows:</p> <ul style="list-style-type: none"> <li>• HA State synchronized—The system is fully synchronized. If a failover becomes necessary, it is non-disruptive.</li> <li>• HA State not in sync—The system is unable to synchronize the two management modules. This may be caused by one or more of the following conditions: <ul style="list-style-type: none"> <li>- A failover is in process but not completed.</li> <li>- The standby management module is faulty.</li> <li>- A system error occurred.</li> </ul> </li> </ul>

## Examples

To display HA status:

```
switch# show ha
SW/1: Active, Warm Recovered, Dual Partitions, Redundant, State synchronized
SW/0: Standby, Dual Partitions, Redundant, State synchronized
```

To display HA status for all partitions:

```
switch# show ha all-partitions
Local (M2): Active, Cold Recovered
Remote (M1): Standby, Healthy
HA enabled, Heartbeat Up, HA State synchronized
L2/0: Active, Cold Recovered, Dual Partitions, Redundant, State in sync
  1: Standby, Dual Partitions, Redundant, State in sync
```

## Related Commands

[ha disable](#), [ha enable](#), [ha failover](#)

# show hardware port-group

Displays the port-group information for the device.

## Syntax

```
show hardware port-group
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays different output, for either a linecard or an MOR.

## Examples

Example output for a linecard:

```
device# show hardware port-group
Group No.          Port Number      Current Mode
=====
3/1                10/3/1-3        Density Mode
3/2                10/3/4-6        Performance Mode
3/3                10/3/7-9        Density Mode
```

Example output for an MOR:

```
device# show hardware port-group
Group No.          Port Number      Current Mode
=====
0/1                10/0/97,99,101  100G
0/2                10/0/98,100,102 40
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show hardware connector-group

Displays all of the connector groups and the corresponding flexports in the groups.

## Syntax

```
show hardware connector-group
```

## Modes

Privileged EXEC mode

## Examples

The **show hardware connector-group** command displays the following information:

```
switch# show hardware connector-group
Connector-group  Flexports
1/0/1           1-8
1/0/3           17-24
1/0/5           33-40
1/0/6           41-48
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show hardware-profile

Displays the route table, ternary content-addressable memory (TCAM), and Keep-Alive Protocol (KAP) profiles in the running configuration, as well as the current active profile information and subtype details for each profile type and RBridge ID.

## Syntax

```
show hardware-profile current [ rbridge-id rbridge-id ]
```

```
show hardware-profile kap { default | profile-name } [ rbridge-id rbridge-id ]
```

```
show hardware-profile rbridge-id { all | rbridge-id } [ current | kap { default } | route-table { default | ipv4-max-route | ipv4-max-arp | ipv4-min-v6 | ipv6-max-route | ipv6-max-nd } | tcam { default | dyn-arp-insp | ipv4-acl | ipv4-v6-mcast | ipv4-v6-pbr | ipv4-v6-qos | l2-acl-qos | l2-ipv4-acl } [ openflow ]
```

```
show hardware-profile route-table { default | ipv4-max-route | ipv4-max-arp | ipv4-min-v6 | ipv6-max-route | ipv6-max-nd } [ openflow ]
```

```
show hardware-profile tcam { default | dyn-arp-insp | ipv4-acl | ipv4-v6-mcast | ipv4-v6-pbr | ipv4-v6-qos | l2-acl-qos | l2-ipv4-acl | openflow } [ rbridge-id { all | rbridge-id } ]
```

## Parameters

### current

Displays current active profile information.

### rbridge-id

Specifies an RBridge ID or all RBridge IDs.

*rbridge-id*

Specifies an RBridge ID. Range of valid values is from 1 through 239.

### all

Specifies all RBridge IDs.

### kap

Optimizes hardware resources for KAP profiles, to support hitless failover for the supported protocols.

**custom-profile** *name*

Specifies a custom profile.

**default**

Optimizes basic support for all applications.

### route-table

Specifies hardware resources for route table profiles.

**default**

Specifies IPv4 or IPv6 resources for dual-stack operations.

**ipv4-max-arp**

Specifies resources for the maximum number of IPv4 ARP entries.

**ipv4-max-route**

Specifies resources for the maximum number of IPv4 routes.



**ipv4-min-v6**

Specifies resources for IPv4 routes in dual-stack configurations.

**ipv6-max-route**

Specifies resources for the maximum number of IPv6 routes.

**ipv6-max-nd**

Specifies resources for the maximum number of IPv6 Neighbor Discovery (ND) entries.

**openflow**

Specifies resources for which OpenFlow is enabled.

**tcam**

Specifies hardware resources for TCAM profiles.

**default**

Specifies resources with basic support for all applications.

**dyn-arp-insp**

Specifies resources for which dynamic ARP inspection (DAI) is enabled.

**ipv4-acl**

Specifies resources for IPv4 ACLs.

**ipv4-v6-mcast**

Specifies resources for multicast.

**ipv4-v6-pbr**

Specifies resources for IPv4 and IPv6 ACLs and policy-based routing tables.

**ipv4-v6-qos**

Specifies resources for IPv4 and IPv6 ACLs and QoS.

**l2-acl-qos**

Specifies resources for Layer 2 ACLs and QoS.

**l2-ipv4-acl**

Specifies resources for Layer 2 and IPv4 ACLs.

**openflow**

Specifies resources for which OpenFlow is enabled.

## Modes

Privileged EXEC mode

RBridge ID configuration mode

## Usage Guidelines

Local hardware profile information can be obtained by means of the **current** keyword. The hardware profile details for any online switch in the cluster can be obtained by means of the **rbridge-id** keyword. If an RBridge is not specified, information for the local switch is displayed.

The **show hardware-profile current** command has been expanded to include the TCAM entry numbers currently "in use", in addition to the "maximum" setting currently displayed. If the current in-use entry number is higher than 90 percent of the maximum, an extra asterisk (\*) is displayed to catch your attention. Meanwhile, Network OS provides active monitoring on the

usage of these TCAM entries, so whenever the 90 percent usage threshold is crossed (in either direction, crossing above or dropping below 90 percent), a RASlog entry is generated. Monitoring the usage of TCAM entries only applies to small form-factor switches, such as the Extreme VDX 6740.

#### NOTE

The **ipv4-acl** keyword is supported only on the Extreme VDX 8770 series.

## Examples

The following example shows the use of the **show hardware-profile** command with the **current** keyword, to show the results of a default profile on an Extreme VDX 6740.

```
device# show hardware-profile current
rbridge-id: 2                switch type:    BR-VDX6740
                             current TCAM profile:    DEFAULT
```

	Max	In-use	
L2 Path Select FCoE:	512	256	
MAC ACL Based QoS Policy Entries (Ingress):	512	128	
L2 Path Select FCoE Zones:	0	0	
MAC Security ACL Entries (Ingress):	512	400	
MAC Policy Based forwarding entries:	0	0	
L2 Multicast No Flood Entries (Ingress):	0	0	
L2 OpenFlow (Ingress):	0	0	
IPV4 ACL Based QoS Policy Entries (Ingress):	512	128	
IPV4 Multicast Entries (Ingress):	1024	800	
IPV4 Policy Based Routing Entries (Ingress):	512	512	*
IPV4 Security ACL Entries (Ingress):	512	400	
L3 Path Select FCoE:	512	480	*
IPV6 Policy Based Routing Entries (Ingress):	0	0	
IPV6 ACL Based QoS Policy Entries (Ingress):	0	0	
IPV6 Multicast Entries (Ingress):	0	0	
IPV6 Security ACL Entries (Ingress):	512	256	
IPV4 OpenFlow (Ingress):	0	0	
IPV4 MINI Path Select Mcast:	0	0	
L3 ACL MINI FCOE:	0	0	
L2 Forward FCoE:	2048	480	
MAC Security ACL Entries (Egress):	128	100	
MAC ACL Based QoS Policy Entries (Egress):	128	100	
IPV4 Security ACL Entries (Egress):	128	100	
IPV4 ACL Based QoS Policy Entries (Egress):	0	0	
IPV6 Security ACL Entries (Egress):	128	100	
IPV6 ACL Based QoS Policy Entries (Egress):	0	0	
FCoE Egress ACL:	128	100	
L2 MAC Classifier:	256	0	
L2 MAC Classifier Prio:	0	0	
VLN Classifier:	4096	0	
Policy Classifier:	0	0	

\*: Exceeded 90% Utilization

The following example displays specific route-table information about resource allocation, facilitating management.

```
device# show hardware-profile route-table ipv4-max-arp openflow
rbridge-id: 200                switch type:    BR-VDX6740
                               route table profile:  IPV4-MAX-ARP-OPENFLOW
-----
                               ipv4_routes:          8096
                               max_nexthops:         768
                               ipv6_routes:           0
                               ipv4_neighbor_cache:    15360
                               ipv6_neighbor_cache:    0
                               fcoe_domain_routes:     2048
-----
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.0	This command was modified to include VLAN classification profiles.
6.0.1	This command was modified to include Keep-Alive Protocol (KAP) profiles, and to remove VLAN classification profiles.
7.0.1	The <b>ipv4-acl</b> keyword was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

# show history

Displays the history of commands executed on the switch.

## Syntax

```
show history [ number ]
```

## Parameters

*number*

The number of commands to display. If you omit this value, all commands are displayed.

## Modes

Privileged EXEC mode

## Examples

Typical command output display.

```
switch# show history
21:10:20 -- show arp vrf test
21:35:57 -- show ip
21:38:03 -- show arp vrf name
21:38:14 -- show access-
21:39:07 -- show access-list-log
21:39:18 -- show capture
21:48:53 -- show b int po
21:48:57 -- show bp
21:53:12 -- show cli
21:53:46 -- show cli
21:54:37 -- show cli
22:05:36 -- show confd-state cli listen ssh ip port
```

# show http server status

Displays the HTTP and HTTPS configuration status of the HTTP Server on each VRF.

## Syntax

```
show http { server status } [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example shows the HTTP and HTTPS server status for all RBridges:

```
show http server status rbridge-id all
rbridge-id 1:
VRF-Name: mgmt-vrf      Status: HTTP Enabled and HTTPS Disabled
VRF-Name: default-vrf  Status: HTTP Enabled and HTTPS Disabled
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	The output of this command was updated.

# show interface

Displays the detailed interface configuration and capabilities of all interfaces or for a specific interface.

## Syntax

```
show interface [ fibrechannel rbridge-id/slot/port | management rbridge-id/slot/port | fcoe [ vn-number/rbridge-id/front-port-number | rbridge-id rbridge-id ] | <N>gigabitethernet rbridge-id/slot/port | loopback number | port-channel number | stats rbridge-id/slot/port | switchport | vlan vlan_id ]
```

## Parameters

### fibrechannel

See **show interface FibreChannel**.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### loopback *number*

Specifies a loopback interface.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### management

See **show interface management**.

### fcoe

Specifies FCoE interfaces.

*vn-number*

Specifies the VN number for FCoE.

*rbridge-id*

Specifies an RBridge ID.

*front-port-number*

Specifies the front port number.

**rbridge-id***rbridge-id*

Specifies an RBridge ID.

**<N> gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel number**

Specifies to display the port-channel number. Valid values range from 1 through 63.

**stats**

See **show interface stats**.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**switchport**

Specifies to display information for Layer 2 interfaces.

**vlan vlan\_id**

Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

If **show interface loopback** is executed in logical chassis cluster mode, loopback interfaces are not shown.

## Examples

The following example displays detailed information for the 10-gigabit Ethernet interface 1/0/1.

```
device# show interface tengigabitethernet 1/0/2

TenGigabitEthernet 1/0/2 is up, line protocol is down (link protocol down)
Hardware is Ethernet, address is 0027.f8cd.1c8d
  Current address is 0027.f8cd.1c8d
Pluggable media not present
Interface index (ifindex) is 4496302080
MTU 2500 bytes
IP MTU 1500 bytes
LineSpeed Actual      : Nil
LineSpeed Configured : Auto, Duplex: Full
Priority Tag disable
Last clearing of show interface counters: 1d17h38m
Queueing strategy: fifo
  Primary Internet Address is 1.1.1.1/32 broadcast is 0.0.0.0
  Donor Interface: TenGigabitEthernet 1/0/1
  IP Unnumbered is enabled
Receive Statistics:
  0 packets, 0 bytes
  Unicasts: 0, Multicasts: 0, Broadcasts: 0
  64-byte pkts: 0, Over 64-byte pkts: 0, Over 127-byte pkts: 0
  Over 255-byte pkts: 0, Over 511-byte pkts: 0, Over 1023-byte pkts: 0
  Over 1518-byte pkts(Jumbo): 0
  Runts: 0, Jabbers: 0, CRC: 0, Overruns: 0
  Errors: 0, Discards: 0
Transmit Statistics:
  0 packets, 0 bytes
  Unicasts: 0, Multicasts: 0, Broadcasts: 0
  Underruns: 0
  Errors: 0, Discards: 0
Rate info:
  Input 0.000000 Mbits/sec, 0 packets/sec, 0.00% of line-rate
  Output 0.000000 Mbits/sec, 0 packets/sec, 0.00% of line-rate
Time since last interface status change: 1d17h38m
```

The following example displays detailed information for a 1-gigabit Ethernet interface.

```
device# show interface gigabitethernet 1/0/2

Gigabit Ethernet 1/0/2 is up, line protocol is up (connected)
Hardware is Ethernet, address is 0005.1e76.1aa5
  Current address is 0005.3313.ac7f
Fixed copper RJ-45 media present
Interface index (ifindex) is 4697661440
MTU 2500 bytes
LineSpeed: 1000 Mbit, Duplex: Full
Flowcontrol rx: off, tx: off
Last clearing of show interface counters: 1d12h37m
Queueing strategy: fifo
Receive Statistics:
  0 packets, 0 bytes
  Unicasts: 0, Multicasts: 0, Broadcasts: 0
  64-byte pkts: 0, Over 64-byte pkts: 0, Over 127-byte pkts: 0
  Over 255-byte pkts: 0, Over 511-byte pkts: 0, Over 1023-byte pkts: 0
  Over 1518-byte pkts(Jumbo): 0
  Runts: 0, Jabbers: 0, CRC: 0, Overruns: 0
  Errors: 0, Discards: 0
Transmit Statistics:
  4425 packets, 513300 bytes
  Unicasts: 4425, Multicasts: 0, Broadcasts: 0
  Underruns: 0
  Errors: 0, Discards: 0
Rate info:
  Input 0.000000 Mbits/sec, 0 packets/sec, 0.00% of line-rate
  Output 0.000000 Mbits/sec, 0 packets/sec, 0.00% of line-rate
Time since last interface status change: 15:14:13
```



The following example displays Layer 2 information for all interfaces.

```
device# show interface switchport

Interface name      : Ten Gigabit Ethernet 1/0/8
Switchport mode    : access
Ingress filter     : enable
Acceptable frame types : all
Default Vlan       : 1
Active Vlans       : 1
Inactive Vlans     : -
Interface name     : Ten Gigabit Ethernet 1/0/19
Switchport mode    : hybrid
Ingress filter     : enable
Acceptable frame types : all
Default Vlan       : 1
Active Vlans       : 1
Inactive Vlans     : 100
Interface name     : Ten Gigabit Ethernet 1/0/20
Switchport mode    : trunk
Ingress filter     : enable
Acceptable frame types : vlan-tagged only
Default Vlan       : 0
Active Vlans       : 1
Inactive Vlans     : -
```

The following example describes an interface that was brought down by link-state tracking (LST) after its uplink was detected as being down.

```
device# show interface tengigabitethernet 3/0/24
TenGigabitEthernet 3/0/24 is admin down, line protocol is down (LST down)

Hardware is Ethernet, address is 0005.3365.3d46
  Current address is 0005.3365.3d46
Pluggable media present, Media type is sfp
  Wavelength is 3072nm
Interface index (ifindex) is 402784259
MTU 2500 bytes
  IPV6 MTU 0 bytes
LineSpeed: Actual Nil Configured Auto, Duplex: Full
Flowcontrol rx: off, tx: off
Last clearing of show interface counters: 00:08:29
Queueing strategy: fifo
Arp ageing timeout value is 04:00:00 (Default)
Receive Statistics:
  24 packets, 2988 bytes
  Unicasts: 0, Multicasts: 24, Broadcasts: 0
  64-byte pkts: 0, Over 64-byte pkts: 6, Over 127-byte pkts: 18
  Over 255-byte pkts: 0, Over 511-byte pkts: 0, Over 1023-byte pkts: 0
  Over 1518-byte pkts(Jumbo): 0
  Runts: 0, Jabbers: 0, CRC: 0, Overruns: 0
  Errors: 0, Discards: 0
Transmit Statistics:
  24 packets, 2968 bytes
```

## History

Release version	Command history
7.0.0	This command was modified to show examples for unnumbered IP interfaces.

# show interface description

Displays the interface description.

## Syntax

```
show interface description [ rbridge-id rbridge-id | range | all ]
```

## Parameters

**rbridge-id** *rbridge-id*

The unique identifier for a switch, or set of switches. The range of valid values is from 1 through 239.

*range*

A range of *rbridge-id* values. The range string can be discontinuous, such as "1-3,5".

**all**

Selects all of the members of the cluster.

## Modes

Privileged EXEC mode

## Examples

Typical command output:

```
device# show interface description
```

```
-----
Port           Type   Speed  Description
-----
Te 1/0/1       eth    10G    appcl12-c9c06-a05-swid1105-m1-sw
Te 1/0/2       eth    10G    --
Te 1/0/3       eth    10G    --
-----
Interface      Description
-----
Po 11          T0:appcl12-c9c06-a06-swid1106-sw-slot104
Po 12          T0:appcl12-c9c06-a06-swid1106-sw-slot105
```

# show interface FibreChannel

Displays the up or down status of the port and of the Fibre Channel protocol, whether pluggable media are present, and additional configuration information.

## Syntax

```
show interface FibreChannel rbridge-id/slot/port [ detail ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**detail**

Displays detailed error and statistics counters.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command applies only to Extreme VDX 6740, VDX 6740T, VDX 6740T-1G, VDX 2741, and VDX 2746 switches.

## Command Output

General **show interface FibreChannel** command configuration information:

Output field	Description
LineSpeed	Actual Operating speed of the port.
PortSpeed	Fixed speed of the port.
portDisableReason	The reason the port is disabled.

Following the general information, the command displays three columns of counters. The first column shows interrupt statistics:

Output field	Description
Interrupts	Total number of interrupts.
Unknown	Interrupts that are not counted elsewhere.
Lli	Low-level interface (physical state, primitive sequences).
Proc_rqrd	Frames delivered for embedded N_Port processing.

Output field	Description
Timed_out	Frames that have timed out.
Rx_flushed	Frames requiring translation.
Tx_unavail	Frames returned from an unavailable transmitter.
Free_buffer	Free buffer available interrupts.
Overrun	Buffer overrun interrupts.
2_parity_err	Secondary transmission parity errors.
Suspended	Transmission suspended interrupts.
Parity_err	Central memory parity errors.

The second column displays link error status block counters:

Output field	Description
Link_failure	Link failures
Loss_of_sync	Synchronization losses
Loss_of_sig	Signal losses
Protocol_err	Protocol errors
Invalid_word	Invalid words
Invalid_crc	Cyclic redundancy errors
Delim_err	Delimiter errors
Address_err	Addressing errors
Lr_in	Line resets in
Li_out	Line resets out
Ols_in	Offline primitive sequences in
Ols_out	Offline primitive sequences out

The third column shows the number of transmitted frames rejected and busied:

Output field	Description
Frjt	Transmitted frames rejected.
Fbsy	Transmitted frames busied.

After this, some transmission rate information is displayed:

Output field	Description
Bandwidth	Bandwidth of the port.
Tx performance	Bytes per second transmitted.
Rx performance	Bytes per second received.

When used with the **detail** parameter, this command also reports the following receive statistics, transmit statistics, error statistics, and port error information.

Output field	Description
stat_wrx	4-byte words received.

Output field	Description
stat_frx	Frames received.
stat_c2_frx	Class 2 frames received.
stat_c3_frx	Class 3 frames received.
stat_lc_rx	Link control frames received.
stat_mc_rx	Multicast frames received.
stat_wtx	4-byte words transmitted.
stat_ftx	Frames transmitted.
stat_mc_tx	Multicast frames transmitted.
tim_txcrd_z	The number of times that the port was unable to transmit frames because the transmit buffer-to-buffer (BB) credit was zero. The purpose of this statistic is to detect congestion or a device affected by latency. This parameter is sampled at intervals of 2.5 microseconds, and the counter is incremented if the condition is true. Each sample represents 2.5 microseconds of time with 0 Tx BB Credit. An increment of this counter means that the frames could not be sent to the attached device for 2.5 microseconds, indicating degraded performance.
er_enc_in	Encoding errors inside frames.
er_crc	Frames with cyclic redundancy check (CRC) errors.
er_trunc	Frames shorter than the minimum frame length.
er_toolong	Frames longer than the maximum frame length.
er_bad_eof	Frames with bad end-of-frame.
er_enc_out	Encoding error outside frames.
er_bad_os	Invalid ordered sets (platform-specific and port-specific).
er_rx_c3_timeout	Receive class 3 frames received at this port and discarded at the transmission port due to timeout (platform-specific and port-specific).
er_tx_c3_timeout	Transmit class 3 frames discarded at the transmission port due to timeout (platform-specific and port-specific).
er_c3_dest_unreach	Class 3 frames discarded because the transmit port, although it is determined, cannot send the frame at the moment when the error occurs.
er_other_discard	Other discards due to route lookup failures or other reasons.
er_type1_miss	FCR frames with transmit errors.
er_type2_miss	Frames with routing errors.
er_type6_miss	FCR frames with receive errors.
er_zone_miss	Frames discarded due to hard zoning miss. Hardware zoning enforcement is not supported currently.
er_lun_zone_miss	Frames discarded due to zoning miss. LUN zoning is not supported currently.
er_crc_good_eof	CRC errors with good end-of-frame (EOF).
er_inv_arb	Invalid ARBs.
Loss_of_sync	Link synchronization errors.
Loss_of_sig	Link loss-of-signal errors.
Frjt	Transmitted frames rejected.
Fbsy	Transmitted frames busied.

## Buffer information:

Output field	Description
Lx Mode	<ul style="list-style-type: none"> <li>L0—Link not in long-distance mode.</li> </ul>

Output field	Description
	<ul style="list-style-type: none"> <li>LD—Link is from 5 km through 10 km.</li> <li>LE—Distance is determined dynamically.</li> <li>LS—Distance is determined statically by user.</li> </ul>
Max/Resv	Buffers The maximum or reserved number of buffers that are allocated to the port based on the estimated distance (as defined by the desire-distance command). If the port is not configured in long distance mode, some systems might reserve buffers for the port. This field then displays the number of buffers reserved for the port.
Buffer Usage	The actual number of buffers allocated to the port. In LD mode, the number is determined by the actual distance and the user-specified desired distance (as defined by the desired-distance command).
Needed Buffers	The number of buffers needed to utilize the port at full bandwidth (depending on the port configuration). If the number of Buffer Usage is less than the number of Needed Buffers, the port is operating in the buffer limited mode.
Link Distance	For LO (not in long distance mode), the command displays the fixed distance based on port speed, for instance: 10 km (1 Gbps), 5 km (2 Gbps), 2 km (4 Gbps), or 1 km (8 Gbps). For static long distance mode (LE), the fixed distance of 10 km displays. For LD mode, Extreme devices use a proprietary algorithm to estimate distance across an ISL. LD mode supports distances up to 500 km. Distance measurement on a link longer than 500 km might not be accurate. If the connecting port does not support LD mode, is shows "N/A".
Remaining Buffers	The remaining (unallocated and reserved) buffers in a port group.

**NOTE**

A hyphen in one of the Buffer information display fields indicates that no relevant information is available; there may be no connection to a port, or the port is disabled, or the port is not an E\_Port.

## Examples

To view Fibre Channel port statistics:

```
device# show interface FibreChannel 66/0/1
fibrechannel 66/0/1 is up. Protocol state is up (connected)
Pluggable media present
LineSpeed Actual: 400,800_MB/s
PortSpeed: N8Gbps
portDisableReason: None
PortId: 427900
PortIfId: 4302303f
PortWwn: 20:79:00:05:33:67:26:78
Distance: normal
Last clearing of show interface counters: 00:00:00
Interrupts: 0 Link_failure: 0 Frjt: 0
Unknown: 0 Loss_of_sync: 1 Fbsy: 0
Lli: 9 Loss_of_sig: 2
Proc_rqrd: 5 Protocol_err: 0
Timed_out: 0 Invalid_word: 0
Rx_flushed: 0 Invalid_crc: 0
Tx_unavail: 0 Delim_err: 0
Free_buffer: 0 Address_err: 0
Overrun: 0 Lr_in: 1
Suspended: 0 Lr_out: 1
Parity_err: 0 Ols_in: 0
2_parity_err: 0 Ols_out: 1
Rate info:
Bandwidth: 8.00G
Tx performance: 0 B/sec
Rx performance: 0 B/sec
```

To view Fibre Channel port statistics details:

```

device# show interface FibreChannel 66/0/1 detail
fibrechannel 66/0/1 is up. Protocol state is up (connected)
Pluggable media present
LineSpeed Actual: 400,800_MB/s
portSpeed: N8Gbps
portDisableReason: None
portId 423100
portIfId: 43020026
portWwn: 20:31:00:05:33:6f:27:57
Distance normal
Last clearing of show interface counters: 00:00:00
Rx Statistics:
stat_wrx 118 4-byte words received
stat_frx 4 Frames received
stat_c2_frx 0 Class 2 frames received
stat_c3_frx 0 Class 3 frames received
stat_lc_rx 2 Link control frames received
stat_mc_rx 0 Multicast frames received
Tx Statistics:
stat_wtx 282 4-byte words transmitted
stat_ftx 12 Frames transmitted
stat_mc_tx 0 Multicast frames transmitted
tim_txcrd_z 2881 Time TX Credit Zero (2.5Us ticks)
tim_txcrd_z_vc 0- 3: 2881 0 0 0
tim_txcrd_z_vc 4- 7: 0 0 0 0
tim_txcrd_z_vc 8-11: 0 0 0 0
tim_txcrd_z_vc 12-15: 0 0 0 0
Error Statistics
er_enc_in 0 Encoding errors inside of frames
er_crc 0 Frames with CRC errors
er_trunc 0 Frames shorter than minimum
er_toolong 0 Frames longer than maximum
er_bad_eof 0 Frames with bad end-of-frame
er_enc_out 0 Encoding error outside of frames
er_bad_os 1 Invalid ordered set
er_rx_c3_timeout 0 Class 3 receive frames discarded due to timeout
er_tx_c3_timeout 0 Class 3 transmit frames discarded due to timeout
er_c3_dest_unreach 0 Class 3 frames discarded due to destination
er_type2_miss 0 frames with FTB Type 2 miss
er_type6_miss 0 frames with FTB type 6 miss
er_zone_miss 0 frames with hard zoning miss
er_lun_zone_miss 0 frames with LUN zoning miss
er_crc_good_eof 0 Crc error with good eof
er_inv_arb 0 Invalid ARB
Port Error Info:
Loss_of_sync:1
Loss_of_sig:2
Frjt:0
Fbsy:0
Buffer Information:
Lx Max/Resv Buffer Needed Link Remaining
Mode Buffers Usage Buffers Distance Buffers
=====
- 8 0 0 - 924
Rate info:
Bandwidth: 8.00G
Tx performance: 0 B/sec
Rx performance: 0 B/sec

```

## Related Commands

[show running-config interface FibreChannel](#)

# show interface management

Displays information related to a management interface.

## Syntax

```
show interface management [ rbridge-id/port ] [ ip [ address | gateway-address ] | ipv6 [ ipv6-address | ipv6-gateways ]  
[ line-speed ] [ oper-status ]
```

## Command Default

This command is executed on the local switch.

## Parameters

*rbridge-id/port*

Specifies the management interface to be configured as the *rbridge-id* followed by a slash (/) and the port number.

*port*

On standalone platforms, the port number for the management port is always 0. On a modular switches with two redundant management modules, you can configure two management ports: 1 and 2.

**ip**

Displays the IPv4 configurations for the selected interface.

**address**

Displays assigned IPv4 addresses.

**gateway-address**

Displays assigned IPv4 gateway addresses.

**ipv6**

Displays the IPv6 configurations for the selected interface.

**ipv6-address**

Displays assigned IPv6 addresses.

**ipv6-gateways**

Displays assigned IPv6 gateway addresses.

**line-speed**

Displays Ethernet speed and other line configurations for the selected interface.

**oper-status**

Lists whether the management interface is up or down.

## Modes

Privileged EXEC mode



## Usage Guidelines

The address field indicates if DHCP is used to obtain an IP address or if a static IP address is used.

## Examples

The following example displays information related to a management interface configured with an IPv4 address:

```
switch# show interface management

Management 2/0
ip address 10.20.49.112/20
ip gateway-address 10.20.48.1
ipv6 ipv6_address [ ]
ipv6 ipv6_gateways [ fe80::21b:edff:fe0b:2400 ]
Linespeed configured "10 Mbit, Duplex: Full"
oper-status up
```

The following example displays information related to a management interface configured with a static IPv6 address:

```
switch# show interface management

interface Management 1/0
ip address 10.17.19.145/20
ip gateway-address 10.17.16.1
ipv6 ipv6-address [ "static aaaa::aaaa/64 preferred" ]
ipv6 ipv6-gateways [ fe80::21b:edff:fe0b:3c00 fe80::21b:edff:fe0b:9000 ]
line-speed actual "1000baseT, Duplex: Full"
line-speed configured Auto
oper-status up
```

The following example displays information related to a management interface on an Extreme VDX 8770. Interface 1/1 is configured with stateless IPv6 addresses:

```
switch# show interface management

interface Management 1/1
ip address 10.24.82.121/20
ip gateway-address 10.24.80.1
ipv6 ipv6-address [ "stateless fd00:60:69bc:64:205:33ff:fe15:f980/64 preferred" ]
ipv6 ipv6-gateways [ fe80::21b:edff:fe0f:bc00 fe80::21b:edff:fe0c:c200 ]
line-speed actual "1000baseT, Duplex: Full"
line-speed configured Auto
interface Management 1/2
ip address 10.24.82.255/20
ip gateway-address 10.24.80.1
ipv6 ipv6-address [ ]
ipv6 ipv6-gateways [ ]
line-speed actual "1000baseT, Duplex: Full"
line-speed configured Auto
oper-status up
```

## Related Commands

[interface management](#), [show running-config interface management](#)

# show interface stats

Displays interface statistics for a variety of interfaces.

## Syntax

```
show interface stats cpu [ rbridge-id { rbridge-id | all } ]
show interface stats brief [ rbridge-id { rbridge-id | all } | slot linecard_number ]
show interface stats detail { rbridge-id { rbridge-id | all } | slot linecard_number } ]
show interface stats detail [ interface { <N>gigabitethernet [ rbridge-id / ] slot / port | port-channel number }
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridges.

### cpu

Displays control packets for the CPU.

### brief

Displays summary statistics.

### slot *linecard\_number*

Displays statistics for specified linecard.

### detail

Displays detailed statistics.

### interface

Displays statistics for a specified interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel number. Valid values range from 1 through 6144.

**slot number**  
Specifies a slot.

## Modes

Privileged EXEC mode

## Examples

The following example displays detailed statistics on a 10-GbE interface.

```
device# show interface stats detail interface ten 1/0/24
Interface TenGigabitEthernet 1/0/24 statistics (ifindex 403439639)
      Packets                RX                TX
      Bytes                  0                0
      Unicasts                0                0
      Multicasts              0                0
      Broadcasts              0                0
      Errors                  0                0
      Discards                0                0
      Overruns                0                Underruns 0
      Runts                   0
      Jabbers                  0
      CRC                     0
      64-byte pkts            0
      Over 64-byte pkts       0
      Over 127-byte pkts      0
      Over 255-byte pkts      0
      Over 511-byte pkts      0
      Over 1023-byte pkts     0
      Over 1518-byte pkts     0
      Mbits/Sec               0.000000        0.000000
      Packet/Sec              0                0
      Line-rate                0.00%           0.00%
```

show interface stats

The following example displays CPU statistics on a specified RBridge.

```
device# show interface stats cpu rbridge-id 8
```

Interface	Packets		Error	
	rx	tx	rx	tx
=====	=====	=====	=====	=====
Fo 8/0/49	0	0	0	0
Fo 8/0/50	0	0	0	0
Fo 8/0/51	0	0	0	0
Fo 8/0/52	0	0	0	0
Te 8/0/1	4924	4924	0	0
Te 8/0/2	9848	9848	0	0
Te 8/0/3	14772	14772	0	0
Te 8/0/4	19696	19696	0	0
Te 8/0/5	24622	24620	0	0
Te 8/0/6	29546	29544	0	0
Te 8/0/7	34470	34468	0	0
Te 8/0/8	39394	39392	0	0
Te 8/0/9	39394	39392	0	0
Te 8/0/10	39394	39392	0	0
Te 8/0/11	39394	39392	0	0
Te 8/0/12	39394	39392	0	0
Te 8/0/13	39394	39392	0	0
Te 8/0/14	39394	39392	0	0
Te 8/0/15	39394	39392	0	0
Te 8/0/16	39394	39392	0	0
Te 8/0/17	44318	44316	0	0
Te 8/0/18	49242	49240	0	0
Te 8/0/19	54166	54164	0	0
Te 8/0/20	59090	59088	0	0
Te 8/0/21	59090	59088	0	0
Te 8/0/22	59090	59088	0	0
Te 8/0/23	59090	59088	0	0
Te 8/0/24	59090	59088	0	0
Te 8/0/25	59090	59088	0	0
Te 8/0/26	59090	59088	0	0
Te 8/0/27	59090	59088	0	0
Te 8/0/28	59090	59088	0	0

## Related Commands

[show interface](#), [show ip interface](#)

# show interface status

Displays the interface status.

## Syntax

```
show interface status [ rbridge-id rbridge-id | range | all ]
```

## Parameters

**rbridge-id** *rbridge-id*

The unique identifier for a switch, or set of switches. The range of valid values is from 1 through 239.

*range*

A range of *rbridge-id* values. The range string can be discontinuous, such as "1-3,5".

**all**

Selects all of the members of the cluster.

## Modes

Privileged EXEC mode

## Examples

Typical command output:

```
device# show interface status
```

Port	Status	Vlan	Speed	Type	Description
Te 1/0/1	connected	Trunk	10G	10G-SFPP-LR	
Te 1/0/2	connected	1	10G	10G-SFPP-SR	
Po 1	connected	Trunk	40G	--	
Po 2	connected	1	20G	--	

# show interface trunk

Displays the interface trunk information.

## Syntax

```
show interface trunk [ rbridge-id rbridge-id | range | all ]
```

## Parameters

**rbridge-id** *rbridge-id*

The unique identifier for a switch, or set of switches. The range of valid values is from 1 through 239.

*range*

A range of *rbridge-id* values. The range string can be discontinuous, such as "1-3,5".

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

Typical command output:

```
device# show interface trunk
```

```
-----  
Port          Vlans Allowed on Trunk  
-----  
Te 1/0/1      1-4094  
Te 1/0/2      1-4094  
Te 1/0/3      1-4094  
Po 52         1-4094  
Po 99         1-4094  
Po 401        701-703,757,2200-2399  
Po 403        701-703,757,2200-2399  
Po 405        701-703,757,2200-2399  
Po 407        701-703,757,2200-2399
```

# show inventory

Displays the hardware inventory of the device.

## Syntax

```
show inventory [ chassis | fan | module | powerSupply ]
```

## Parameters

### chassis

Displays information about the chassis.

### fan

Displays information about the fan.

### module

Displays information about the module.

### powerSupply

Displays information about the power supply.

## Modes

Privileged EXEC mode

## Examples

The following is an example of typical command output.

```
device# show inventory
NAME:MM, Slot M1      DESCR:Chassis Blade module
PN:60-1002179-23     SN:BVT0417J00M
NAME:MM, Slot M2      DESCR:Chassis Blade module
PN:60-1002179-13     SN:BVT0302H00T
NAME:SFM, Slot S1     DESCR:Chassis Blade module
PN:60-1002180-12     SN:BVU0304H037
NAME:SFM, Slot S2     DESCR:Chassis Blade module
PN:60-1002180-12     SN:BVU0302H01Y
NAME:SFM, Slot S3     DESCR:Chassis Blade module
PN:60-1002560-01     SN:BVU0307H01G
NAME:LC, Slot L1      DESCR:Chassis Blade module
PN:60-1002466-17     SN:CCE0423J00P
NAME:LC, Slot L2      DESCR:Chassis Blade module
PN:60-1002466-09     SN:CCE0315H00B
NAME:LC, Slot L3      DESCR:Chassis Blade module
PN:60-1002569-01     SN:CCE0305H00G
NAME:LC, Slot L4      DESCR:Chassis Blade module
PN:60-1002181-12     SN:BVV0303H019
NAME:POWER SUPPLY 1  DESCR:Chassis PS module
PN:23-0000135-02     SN:BMM2J25G998
NAME:POWER SUPPLY 2  DESCR:Chassis PS module
PN:23-0000135-02     SN:BMM2J25G803
NAME: Chassis        DESCR:System Chassis
SID:BR-VDX8770-4     SwitchType:1000
PN:84-1001681-03     SN:BZA0305H00D
```

# show ip anycast-gateway

Displays IPv4 anycast-gateway details for all virtual Ethernet (VEs) or for a specified VE. You can also filter by RBridge and by VRF.

## Syntax

```
show ip anycast-gateway [ interface ve vlan-id ] [ rbridge-id { rbridge-id | range | all } ]
```

```
show ip anycast-gateway [ vrf { vrf-name | all } ] [ rbridge-id { rbridge-id | range | all } ]
```

## Parameters

**interface ve** *vlan-id*

Specifies a virtual Ethernet (VE) interface.

**rbridge-id**

Specifies an RBridge, multiple RBridges, or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

*range*

Specifies multiple RBridge IDs. You can specify a range (for example, 3-5), a comma-separated list (for example, 1,3,5,6), or you can combine a range with a list (for example, 1-5,6,8). In a range string, no spaces are allowed.

**all**

Specifies all RBridges.

**vrf**

Specifies all VRF instances or one VRF instance. Without the **vrf** keyword, details for the default VRF instance are shown in the output.

*vrf-name*

Specifies a vrf name.

**all**

Specifies all vrfs.

## Modes

Privileged EXEC mode

## Command Output

The **show ip anycast-gateway** command displays the following information:

Output field	Description
Gateway mac	Displays the MAC address specified for the anycast gateway.
Interface	Displays the virtual Ethernet (VE) interface.



Output field	Description
ip address	Displays the IP anycast address and mask.
state	Displays "Active" or "Inactive". If the value is "Inactive", the reason is displayed in brackets.

## Examples

The following example displays IPv4 anycast-gateway information on all VEs.

```
device# show ip anycast-gateway
Gateway mac: 000a.000b.000c
Interface      Ip address      state
ve10           2.2.2.2/24      Active
ve10           2.2.3.2/24      Active
ve20           3.3.3.3/24      Active
```

The following example displays IPv4 anycast-gateway information on a specified VE.

```
device# show ip anycast-gateway interface ve 10
Gateway mac: 000a.000b.000c
Interface      Ip address      state
ve10           2.2.2.2/24      Active
ve10           2.2.3.2/24      Active
```

The following example displays the default vrf anycast-gateway sessions. (This option is equivalent to the **show ip anycast-gateway** command with no additional parameters.)

```
show ip anycast-gateway vrf default-vrf
Interface      Ip address      state
Gateway mac: 0000.aaaa.cccc

Ve15           2.2.2.2/24      Active
Ve16           2.2.3.2/24      Active
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show ip arp inspection

Displays dynamic ARP inspection (DAI) information for one or more VLANs.

## Syntax

```
show ip arp inspection [ vlan vlan-range ]
```

## Parameters

*vlan-range*

Specifies a VLAN, multiple VLANs (separated by commas with no spaces), a range of VLANs, or a combination of specified VLANs and ranges of VLANs. If a virtual fabric is disabled, valid values are 1 through 4090. If a virtual fabric is enabled, valid values are 1 through 8191.

## Modes

Privileged EXEC mode

## Command Output

The **show ip arp inspection** command displays the following information:

Output field	Description
Vlan	Displays the VLAN name.
Configuration	Displays Enabled ( <b>ip arp inspection</b> ) or Disabled ( <b>no ip arp inspection</b> ).
Operation	Displays "Active" if ARP configuration is successfully saved to the database.  "Inactive" indicates one of the following conditions: <ul style="list-style-type: none"> <li>The "Configuration" value is "Disabled".</li> <li>There is an internal issue that prevents successful application of ACLs</li> </ul> Displays "Partial" if ARP ACLs are successfully applied to one or more linecards/RBridges but not to all of them
ACL Match	Displays the name of the ARP ACL that is applied.
ACL Logging	Displays "matchlog" if <b>arp inspection logging acl-match</b> was entered (logging is enabled).  Does not display a value if logging is either disabled ( <b>arp inspection logging acl-match none</b> ) or not configured.

## Examples

The following example displays DAI information for all VLANs.

```
device# show ip arp inspection
Vlan  Configuraton  Operation  ACL Match  ACL Logging
-----
   1      Enabled      Active
  10     Disabled     Inactive
 100     Enabled      Active      ac11      Acl-Match
1000     Enabled      Active      Acl-Match
   20     Disabled     Inactive
  200     Disabled     Inactive      Acl-Match
 2000     Enabled      Active      ac11
```

The following example displays DAI information for specified VLANs and a range of VLANs.

```
device# show ip arp inspection vlan 1,100,200-2000
Vlan  Configuraton  Operation  ACL Match  ACL Logging
-----
   1      Enabled      Active
  100     Enabled      Active      ac11      Acl-Match
1000     Enabled      Active      Acl-Match
  200     Disabled     Inactive      Acl-Match
 2000     Enabled      Active      ac11
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show ip arp inspection interfaces

For VLANs enabled for dynamic ARP inspection (DAI), displays a list of trusted interfaces.

## Syntax

```
show ip arp inspection interfaces [ <N>gigabitethernet rbridge_id/slot/port | port-channel index ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tenGigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge\_id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### port-channel index

Specifies a port-channel interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

For VLANs enabled for dynamic ARP inspection (DAI), interfaces not listed by this command are untrusted.

## Command Output

The **show ip arp inspection interfaces** command displays the following information:

Output field	Description
Interface	Displays a prefix specifying the interface type, followed by the interface identifier.
Trust State	Displays "Trusted".

## Examples

The following example displays all trusted interfaces.

```
device# show ip arp inspection interfaces
Interface      Trust State
-----
Po 200         Trusted
Te 2/0/9       Trusted
Te 4/0/10      Trusted
-----
```

All other interfaces are untrusted.

The following example displays the trust state of TenGigabitEthernet interface 4/0/10.

```
device# show ip arp inspection interfaces tengigabitethernet 4/0/10
Interface      Trust State
-----
Te 4/0/10      Trusted
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show ip arp inspection statistics

Displays dynamic ARP inspection (DAI) statistics for one or more DAI-enabled VLANs.

## Syntax

```
show ip arp inspection statistics [ vlan vlan-range ]
```

## Parameters

**vlan** *vlan-range*

Specifies a VLAN, multiple VLANs (separated by commas with no spaces), a range of VLANs, or a combination of specified VLANs and ranges of VLANs. If a virtual fabric is disabled, valid values are 1 through 4090. If a virtual fabric is enabled, valid values are 1 through 8191.

## Modes

Privileged EXEC mode

## Command Output

The **show ip arp inspection statistics** command displays the following information:

Output field	Description
Vlan	Displays the VLAN name.
Forwarded	Displays packets forwarded, included packets permitted by ARP ACLs.
Dropped	Displays packets dropped as a result of DAI policy.
ACL Permit	Displays packets forwarded based on <b>permit</b> statements in ARP ACLs.

## Examples

The following example displays statistics for VLAN 400.

```
device# sshow ip arp inspection statistics vlan 400
  Vlan      Forwarded      Dropped      ACL Permit
-----
  400              0              0              0
```

The following example displays statistics for all DAI-enabled VLANs.

```
device# show ip arp inspection statistics
  Vlan      Forwarded      Dropped      ACL Permit
-----
  1              0              0              0
  2              0              0              0
  3      68322      220356      68322
  4              0              0              0
  100         0              0              0
  101         0              0              0
  1006        0              0              0
  1007        0              0              0
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show ip arp suppression-cache

Displays IPv4 ARP-suppression information.

## Syntax

```
show ip arp suppression-cache
```

```
show ip arp suppression-cache [ summary | vlan vlan-id ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### summary

Specifies summary format.

### vlan *vlan-id*

Specifies a VLAN interface.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the results of the basic form of this command.

```
device# show ip arp suppression-cache
Flags: L - Local Learnt Adjacency
       R - Remote Learnt Adjacency
       RS - Remote Static Adjacency
```

Vlan	IP	Mac	Interface	Age	Flags
Ve 100	1.100.1.32	50eb.1aaa.30b1	Te 100/0/49:2	00:00:05	L
Ve 100	1.100.1.33	50eb.1aac.2c41	Tu 61441	Never	RS
Ve 254	1.254.1.201	50eb.1aaa.30b1	Tu 61441	Never	R
Ve 254	1.254.1.202	50eb.1aac.2c41	Tu 61441	Never	RS

(Output truncated for brevity)

## History

Release version	Command history
7.0.0	This command was introduced.



# show ip arp suppression-statistics

Displays IPv4 ARP-suppression statistics.

## Syntax

```
show ip arp suppression-statistics [ vlan vlan-id ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

**vlan** *vlan-id*

Specifies a VLAN interface.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the results of the basic form of this command.

```
device# show ip arp suppression-statistics
Vlan      Forwarded   Suppressed  Remote-arp Proxy
-----
110       0           24          0
254       3           10          0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show ip arp suppression-status

Displays the IPv4 ARP-suppression status.

## Syntax

```
show ip arp suppression-status [ vlan vlan-id ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

**vlan** *vlan-id*

Specifies a VLAN interface.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the results of the basic form of this command.

```
device# show ip arp suppression-status
Vlan      Configuration  Evpn-Register  Operation
-----
1          Disabled      No              Inactive
100       Disabled      No              Inactive
110       Enabled       Yes             Active
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show ip as-path-list

Displays the status of BGP AS-path access control lists (ACLs).

## Syntax

```
show ip as-path-list [ list_name [rbridge-id number] | rbridge-id list_name ]
```

## Parameters

*list\_name*

Name of an Autonomous System (AS) ACL.

*rbridge-idnumber*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To view AS-path ACL status for a specific list:

```
device# show ip as-path-list myaspathlist
```

# show ip bgp

Displays BGP4 route information.

## Syntax

```
show ip bgp ip-addr [ /prefix ]
```

```
show ip bgp { ip-addr [ /prefix ] } [ longer-prefixes | rbridge-id { rbridge-id | all } ] vrf vrf-name ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation, with optional mask.

*/prefix*

IPv4 mask length in CIDR notation.

### **longer-prefixes**

Filters on prefixes equal to or greater than that specified by *prefix*.

### **rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example displays sample output from the **show ip bgp** command.

```
device# show ip bgp
Total number of BGP Routes: 14
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S
stale
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop          MED      LocPrf    Weight Path
*> 10.2.2.0/24      101.101.2.1      1         100       0       101
*> 10.11.11.1/32   101.101.2.1      0         100       0       101
*> 10.1.222.2/31   101.101.2.1      0         100       0       101
*> 10.122.125.0/24 101.101.2.1      0         100       0       101
*> 10.101.1.0/24   101.101.2.1      0         100       0       101
*> 10.11.122.0/24  101.101.2.1      0         100       0       101
*> 10.101.2.0/31   101.101.2.1      0         100       0       101
*> 10.102.10.0/24  101.101.2.1      0         100       0       101
*> 10.102.88.0/24  101.101.2.1      0         100       0       101
*> 10.128.1.0/31   101.101.2.1      0         100       0       101
*> 10.222.10.0/24  101.101.2.1      0         100       0       101
*> 10.25.222.2/31  101.101.2.1      0         100       0       101
*> 10.122.125.0/24 101.101.2.1      1         100       0       101
*> 10.123.124.125/32 101.101.2.1      1         100       0       101
```

This example displays sample output from the **show ip bgp** command when the **vrf** keyword is used and a VRF specified.

```
device# show ip bgp vrf red
Total number of BGP Routes: 4
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S
stale
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop          RD          MED      LocPrf    Weight Path
*> 10.1.1.0/31      0.0.0.0           0           0         100       32768 ?
*> 10.3.1.0/31      0.0.0.0           0           0         100       32768 ?
*> 10.5.1.0/31      0.0.0.0           0           0         100       32768 ?
*> 10.6.1.254/31    0.0.0.0           0           0         100       32768 ?
```

## History

Release version	Command history
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ip bgp attribute-entries

Displays BGP4 route-attribute entries that are stored in device memory.

## Syntax

```
show ip bgp attribute-entries [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

The route-attribute entries table lists the sets of BGP4 attributes that are stored in device memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the device typically has fewer attribute entries than routes. Use this command to view BGP4 route-attribute entries that are stored in device memory.

## Examples

This example shows sample output for the **show ip bgp attribute-entries** command.

```
device# show ip bgp attribute-entries
Total number of BGP Attribute Entries: 2
1   Next Hop : 10.101.2.1      MED      :0           Origin:INCOMP
    Originator:0.0.0.0        Cluster List:None
    Aggregator:AS Number :0    Router-ID:0.0.0.0   Atomic:Non
e
    Local Pref:100           Communities:Internet
    AS Path   :101 (length 3)
    AsPathLen: 1 AsNum: 1, SegmentNum: 1, Neighboring As: 101, Source As
101
    Address: 0x12320a0a Hash:3432 (0x03000296)
    Links: 0x00000000, 0x00000000
    Reference Counts: 11:0:55, Magic: 2
2   Next Hop : 10.101.2.1      MED      :1           Origin:INCOMP
    Originator:0.0.0.0        Cluster List:None
    Aggregator:AS Number :0    Router-ID:0.0.0.0   Atomic:Non
e
    Local Pref:100           Communities:Internet
    AS Path   :101 (length 3)
    AsPathLen: 1 AsNum: 1, SegmentNum: 1, Neighboring As: 101, Source As
101
    Address: 0x123209a4 Hash:3433 (0x03000296)
    Links: 0x00000000, 0x00000000
    Reference Counts: 3:0:15, Magic: 1
```

This example displays route-attribute entries for VRF instance "red".

```
device# show ip bgp attribute-entries vrf red
Total number of BGP Attribute Entries: 1
1   Next Hop : 10.5.5.6        MED      :0           Origin:INCOMP
    Originator:0.0.0.0        Cluster List:None
    Aggregator:AS Number :0    Router-ID:0.0.0.0   Atomic:None
    Local Pref:100           Communities:1:1 2:2 3:3 4:4
    AS Path   : (length 0)
    Address: 0x2699a024 Hash:4491 (0x03000000)
    Links: 0x00000000, 0x00000000
    Reference Counts: 3:0:0, Magic: 8
```

## History

Release version	Command history
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ip bgp dampened-paths

Displays Multiprotocol BGP (MBGP) paths that have been dampened by route-flap dampening.

## Syntax

```
show ip bgp dampened-paths [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridges.

### vrf *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example shows sample output from the **show ip bgp dampened-paths** command when the **vrf** keyword has been used and a VRF specified.

```
device# show ip bgp dampened-paths vrf red
Status Code >:best d:damped h:history *:valid
Network      From      Flp      Since  Reuse      Pnlty rIdx dBlk
*d 5.5.5.0/24 5.5.5.6    5 0 :2 :31 0 :41:50 3590  3  0
*d 6.6.6.6/32 5.5.5.6    5 0 :2 :31 0 :41:50 3590  3  0
```

## History

Release version	Command history
6.0.1	The <b>vrf <i>vrf-name</i></b> parameter was added to support Multi-VRF.



# show ip bgp filtered-routes

Displays BGP4 filtered routes that are received from a neighbor or peer group.

## Syntax

```
show ip bgp filtered-routes [ detail ] [ ip-addr { / mask } [ longer-prefixes ] ] [ as-path-access-list name ] [ prefix-list name ]
[ rbridge-id { rbridge-id | all } ] [ vrf vrf-name ]
```

## Parameters

### detail

Optionally displays detailed route information.

### ip-addr

IPv4 address of the destination network in dotted-decimal notation.

### mask

(Optional) IPv4 mask of the destination network in CIDR notation.

### longer-prefixes

Specifies all statistics for routes that match the specified route, or that have a longer prefix than the specified route.

### as-path-access-list

Specifies an AS-path ACL.

### prefix-list

Specifies an IP prefix list.

### name

Name of an AS-path ACL or prefix list.

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

### vrf vrf-name

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example displays received filtered routes for the prefix list "myprefixlist" for VRF instance "red".

```
device# show ip bgp filtered-routes detail 10.11.12.13 prefix-list myprefixlist vrf red
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp flap-statistics

Displays BGP4 route-dampening statistics for all dampened routes with a variety of options.

## Syntax

**show ip bgp flap-statistics**

**show ip bgp flap-statistics** *ip-addr* { / *mask* } [ **longer-prefixes** [ **rbridge-id** { *rbridge-id* | **all** } ] [ **vrf** *vrf-name* [ **rbridge-id** { *rbridge-id* | **all** } ] ] | **rbridge-id** { *rbridge-id* | **all** } **vrf** *vrf-name* [ **rbridge-id** { *rbridge-id* | **all** } ] ]

**show ip bgp flap-statistics neighbor** *ip-addr* [ **rbridge-id** { *rbridge-id* | **all** } ] [ **vrf** *vrf-name* [ **rbridge-id** { *rbridge-id* | **all** } ] ]

**show ip bgp flap-statistics rbridge-id** { *rbridge-id* | **all** }

**show ip bgp flap-statistics regular-expression** [ **rbridge-id** { *rbridge-id* | **all** } ] [ **vrf** *vrf-name* [ **rbridge-id** { *rbridge-id* | **all** } ] ]

**show ip bgp flap-statistics vrf** *vrf-name* [ **rbridge-id** { *rbridge-id* | **all** } ]

## Parameters

*ip-addr*

IPv4 address of a specified route in dotted-decimal notation.

*mask*

(Optional) IPv4 mask of a specified route in CIDR notation.

**longer-prefixes**

Displays statistics for routes that match the specified route or have a longer prefix than the specified route.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

**neighbor**

Displays flap statistics only for routes learned from the specified neighbor.

*ip-addr*

IPv4 address of the neighbor.

**regular-expression**

Specifies a regular expression in the display output on which to filter.

*name*

Name of an AS-path filter or regular expression.

show ip bgp flap-statistics

## Modes

Privileged EXEC mode

## Examples

This example displays flap statistics for a neighbor.

```
device# show ip bgp flap-statistics neighbor 10.11.12.13
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp neighbors

Displays configuration information and statistics for BGP4 neighbors of the device.

## Syntax

```
show ip bgp neighbors [ ip-addr ]
show ip bgp neighbors last-packet-with-error [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
show ip bgp neighbors routes-summary [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
show ip bgp neighbors rbridge-id { rbridge-id | all }
show ip bgp neighbors vrf vrf-name
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor in dotted-decimal notation.

**last-packet-with-error**

Displays the last packet with an error.

**route-summary**

Displays routes received, routes accepted, number of routes advertised by peer, and so on.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to view configuration information and statistics for BGP4 neighbors of the device. Output shows all configured parameters for the neighbors. Only the parameters whose values differ from defaults are shown. Information about dynamically created BGP neighbors is displayed for this command only when the *ip-addr* parameter is used.

## Examples

The following example shows sample output from the **show ip bgp neighbors** command, including BFD configuration information.

```
device# show ip bgp neighbors

Total number of BGP Neighbors: 1
1  IP Address: 100.1.1.2, AS: 100 (IBGP), RouterID: 19.19.19.19, VRF: default-vrf
   State: ESTABLISHED, Time: 0h6m49s, KeepAliveTime: 60, HoldTime: 180
      KeepAliveTimer Expire in 44 seconds, HoldTimer Expire in 136 seconds
   Minimal Route Advertisement Interval: 0 seconds
      RefreshCapability: Received
Messages:   Open   Update   KeepAlive   Notification   Refresh-Req
Sent       : 1     0         9           0               0
Received: 1     0         8           0               0
Last Update Time: NLRI           Withdraw           NLRI           Withdraw
                  Tx: ---         ---               Rx: ---         ---
Last Connection Reset Reason:Unknown
Notification Sent:      Unspecified
Notification Received: Unspecified
Neighbor NLRI Negotiation:
  Peer Negotiated IPV4 unicast capability
  Peer configured for IPV4 unicast Routes
Neighbor ipv6 MPLS Label Capability Negotiation:
Neighbor AS4 Capability Negotiation:
Outbound Policy Group:
  ID: 4, Use Count: 1
BFD: Enabled,BFDSessionState:UP,Multihop:No
  LastBGP-BFDEvent:RX:Up,BGP-BFDError:No Error
  HoldOverTime(sec) Configured:0,Current:0,DownCount:0
TCP Connection state: ESTABLISHED, flags:00000033 (0,0)
Maximum segment size: 1460
TTL check: 0, value: 0, rcvd: 64
  Byte Sent: 216, Received: 197
  Local host: 100.1.1.1, Local Port: 8177
  Remote host: 100.1.1.2, Remote Port: 179
```

The following example shows sample output from the **show ip bgp neighbors** command, including BGP add path configuration information.

```
device# show ip bgp neighbors

Total number of BGP Neighbors: 4
1  IP Address: 10.0.0.2, AS: 1 (IBGP), RouterID: 10.0.0.2, VRF: default-vrf
   State: ESTABLISHED, Time: 0h2m12s, KeepAliveTime: 60, HoldTime: 180
      KeepAliveTimer Expire in 25 seconds, HoldTimer Expire in 136 seconds
   Minimal Route Advertisement Interval: 0 seconds
      RefreshCapability: Received
      AddPathCapability Sent To Neighbor: Send, Receive
      AddPathCapability Received From Neighbor: Send, Receive
Messages:   Open   Update   KeepAlive   Notification   Refresh-Req
Sent       : 4     8        13          0               0
Received: 4     0        11          3               0
Last Update Time: NLRI           Withdraw           NLRI           Withdraw
                  TX: 0h1m21s         0h1m27s           Rx: ---         ---
Last Connection Reset Reason: Rcv Notification
Notification Sent:      Unspecified
Notification Received: Cease/Administrative Reset
Neighbor NLRI Negotiation:
  Peer configured for IPV4 unicast Routes
Neighbor ipv6 MPLS Label Capability Negotiation:
Neighbor AS4 Capability Negotiation:
Outbound Policy Group:
  ID: 2, Use Count: 4
BFD: Disabled
  Byte Sent: 426, Received: 0
  Local host: 10.0.0.1, Local Port: 179
  Remote host: 10.0.0.2, Remote Port: 8065
```

The following example shows sample output from the **show ip bgp neighbors** command, including BGP time to live (TTL) security hack protection (BTSH) configuration information.

```
device# show ip bgp neighbors 10.11.11.2

1  IP Address: 10.11.11.2, AS: 2 (EBGP), RouterID: 2.2.2.2, VRF: default-vrf
   State: ESTABLISHED, Time: 0h0m28s, KeepAliveTime: 60, HoldTime: 180
     KeepAliveTimer Expire in 23 seconds, HoldTimer Expire in 151 seconds
   Minimal Route Advertisement Interval: 0 seconds
     Multihop-BTSH: yes
     RefreshCapability: Received
   Messages:   Open      Update  KeepAlive  Notification  Refresh-Req
     Sent      : 1        0        1          0              0
     Received: 1        0        1          0              0
   Last Update Time: NLRI                               Withdraw      NLRI                               Withdraw
                   Tx: ---                             ---           Rx: ---      ---
...
BFD:Disabled
  Byte Sent: 64, Received: 0
  Local host: 10.11.11.1, Local Port: 8036
  Remote host: 10.11.11.2, Remote Port: 179
```

The following example shows sample output from the **show ip bgp neighbors** command, including information about dynamically created BGP neighbors.

```
device# show ip bgp neighbors

1  IP Address: 98.0.0.1, AS: 100 (IBGP), RouterID: 98.0.0.1, VRF: default-vrf
   State: ESTABLISHED, Time: 1h56m21s, KeepAliveTime: 60, HoldTime: 180
     KeepAliveTimer Expire in 40 seconds, HoldTimer Expire in 157 seconds
   Minimal Route Advertisement Interval: 0 seconds
     PeerGroup: pgl
       DYNAMIC neighbor belongs to the subnet range group:
         98.0.0.0/24
     RefreshCapability: Received
   Address Family : L2VPN EVPN
     SendExtendedCommunity: yes
   Messages:   Open      Update  KeepAlive  Notification  Refresh-Req
     Sent      : 4        18      178        3              0
     Received: 4        42      177        0              0
   Last Update Time: NLRI                               Withdraw      NLRI                               Withdraw
                   Tx: 1h56m21s                         ---           Rx: 1h56m20s  ---
   Last Connection Reset Reason: User Reset Peer Session
   Notification Sent: Cease/Administrative Reset
   Notification Received: Unspecified
   Neighbor NLRI Negotiation:
     Peer Negotiated L2VPN EVPN address family
     Peer configured for IPV4 unicast Routes
     Peer configured for L2VPN EVPN address family
   Neighbor ipv6 MPLS Label Capability Negotiation:
   Neighbor AS4 Capability Negotiation:
   Outbound Policy Group:
     ID: 2, Use Count: 3
     Last update time was 4667 sec ago
   BFD:Disabled
     Byte Sent: 2968, Received: 0
     Local host: 98.0.0.2, Local Port: 8231
     Remote host: 98.0.0.1, Remote Port: 179
```

The following example shows sample output from the **show ip bgp neighbors** command when an IP address is specified, including information about dynamically created BGP neighbors.

```
device# show ip bgp neighbors 33.11.11.11

1  IP Address: 33.11.11.11, AS: 1234 (IBGP), RouterID: 59.59.59.59, VRF: default-vrf
   State: ESTABLISHED, Time: 1dlh0m13s, KeepAliveTime: 60, HoldTime: 180
     KeepAliveTimer Expire in 41 seconds, HoldTimer Expire in 145 seconds
   Minimal Route Advertisement Interval: 0 seconds
     PeerGroup: ibgp-mh
       DYNAMIC neighbor belongs to the subnet range group:
         33.0.0.0/8
     Updatesource: Loopback 5
     SoftInboundReconfiguration: yes
     RefreshCapability: Received
     GracefulRestartCapability: Received
       Restart Time 120 sec, Restart bit 0
         afi/safi 1/1, Forwarding bit 0
         afi/safi 25/70, Forwarding bit 0
     GracefulRestartCapability: Sent
       Restart Time 120 sec, Restart bit 1
         afi/safi 1/1, Forwarding bit 1
         afi/safi 25/70, Forwarding bit 1
   Address Family : IPV4 Unicast
     RouteReflectorClient: yes
     MaximumPrefixLimit: 100
   Address Family : L2VPN EVPN
     RouteReflectorClient: yes
     SendExtendedCommunity: yes
     MaximumPrefixLimit: 10000
   Messages:
     Open      Update  KeepAlive  Notification  Refresh-Req
     Sent      : 1      1222      1687        0              0
     Received: 1      657      1683        0              1
   Last Update Time: NLRI          Withdraw          NLRI          Withdraw
                   Tx: 13h34m36s      ---              Rx: 13h28m45s      13h28m56s
   Last Connection Reset Reason:Unknown
   Notification Sent:      Unspecified
   Notification Received: Unspecified
   Neighbor NLRI Negotiation:
     Peer Negotiated L2VPN EVPN address family
     Peer configured for IPV4 unicast Routes
     Peer configured for L2VPN EVPN address family
   Neighbor ipv6 MPLS Label Capability Negotiation:
   Neighbor AS4 Capability Negotiation:
     Peer Negotiated AS4 capability
     Peer configured for AS4 capability
   Outbound Policy Group:
     ID: 5, Use Count: 4
   BFD:Disabled
     Byte Sent: 208366, Received: 0
     Local host: 77.77.77.77, Local Port: 179
     Remote host: 33.11.11.11, Remote Port: 8031
```



The following example shows sample output from the **show ip bgp neighbors** command including graceful shutdown configurations.

```
device# show ip bgp neighbors

Total number of BGP Neighbors: 1
1  IP Address: 10.1.1.2, AS: 100 (IBGP), RouterID: 10.1.1.2, VRF: default-vrf
   State: ESTABLISHED, Time: 0h17m22s, KeepAliveTime: 60, HoldTime: 180
     KeepAliveTimer Expire in 13 seconds, HoldTimer Expire in 166 seconds
   Minimal Route Advertisement Interval: 0 seconds
     RefreshCapability: Received
     SendExtendedCommunity: yes
   Messages:   Open   Update   KeepAlive   Notification   Refresh-Req
     Sent      : 1     0         20          0              0
     Received: 1     0         20          0              0
   Last Update Time: NLRI                               Withdraw      NLRI             Withdraw
                   Tx: ---                               ---           Rx: ---         ---
   Last Connection Reset Reason: Peer Removed
   Notification Sent:      Cease/Peer Unconfigured
   Notification Received: Unspecified
   Neighbor NLRI Negotiation:
     Peer Negotiated IPV4 unicast capability
     Peer configured for IPV4 unicast Routes
   Neighbor AS4 Capability Negotiation:
   Outbound Policy Group:
     ID: 4, Use Count: 1
   BFD: Disabled
     Byte Sent: 425, Received: 0
     Local host: 10.1.1.1, Local Port: 8122
     Remote host: 10.1.1.2, Remote Port: 179
   G-Shut:
     Enabled: yes, g-shut timer: 600 seconds, Route-map: none
     g-shut timer Expire in 200 seconds
     local-preference 0 gshut community 1200
```

The following example shows sample output from the **show ip bgp neighbors** command including BGP automatic neighbor discovery configurations.

```
device# show ip bgp neighbors

Total number of BGP Neighbors: 1
1  IP Address: 23.23.23.3, AS: 30 (EBGP), RouterID: 23.23.23.3, VRF: default-vrf
   State: ESTABLISHED, Time: 19h11m58s, KeepAliveTime: 60, HoldTime: 180
     KeepAliveTimer Expire in 18 seconds, HoldTimer Expire in 162 seconds
   Minimal Route Advertisement Interval: 0 seconds
   PeerGroup: grp
     Auto Discovered LLDP Neighbor
     MD5 Password: $M1VzZCFAbg==
     RefreshCapability: Received
   Messages:   Open   Update   KeepAlive   Notification   Refresh-Req
     Sent      : 1     0        1300        0              0
     Received: 1     0        1300        0              0
   Last Update Time: NLRI                               Withdraw      NLRI             Withdraw
                   Tx: ---                               ---           Rx: ---         ---
   Last Connection Reset Reason: Unknown
   Notification Sent:      Unspecified
   Notification Received: Unspecified
   Neighbor NLRI Negotiation:
     Peer configured for IPV4 unicast Routes
   Neighbor ipv6 MPLS Label Capability Negotiation:
   Neighbor AS4 Capability Negotiation:
   Outbound Policy Group:
     ID: 2, Use Count: 1
   BFD: Disabled
     Byte Sent: 24745, Received: 0
     Local host: 23.23.23.2, Local Port: 8164
     Remote host: 23.23.23.3, Remote Port: 179
   G-Shut: Disabled
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF. This command was modified to include BFD configuration status.
7.0.0	This command was modified to include BGP add path configuration status and BGP BTSH configuration status.
7.1.0	This command was modified to display information about dynamically created BGP neighbors for the <b>show ip bgp neighbors</b> command and for the <b>show ip bgp neighbors ip-addr</b> command.
7.2.0	This command was modified to include BGP graceful shutdown and BGP automatic neighbor discovery information.

## Related Commands

[bfd interval](#), [bfd holdover-interval](#), [neighbor bfd](#)

# show ip bgp neighbors advertised-routes

Displays only the routes that the device has advertised to the neighbor during the current BGP4 session.

## Syntax

```
show ip bgp neighbors ip-addr advertised-routes [ detail | ip-addr { / mask-bits } ] [ rbridge-id { rbridge-id | all } ] vrf vrf-name ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**detail**

Displays details of advertised routes.

*mask-bits*

Number of mask bits in CIDR notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example displays the details of advertised routes.

```

device# show ip bgp neighbors 10.154.154.154 advertised-routes
      There are 14 routes advertised to neighbor 10.154.154.154
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
      Prefix          Next Hop      MED          LocPrf      Weight  Status
  1    10.11.11.1/32   10.101.2.1    0            100         0       BE
      AS_PATH: 101
  2    10.1.222.2/31  10.101.2.1    0            100         0       BE
      AS_PATH: 101
  3    10.122.125.0/24 10.101.2.1    0            100         0       BE
      AS_PATH: 101
  4    10.101.1.0/24   10.101.2.1    0            100         0       BE
      AS_PATH: 101
  5    10.11.122.0/24  10.101.2.1    0            100         0       BE
      AS_PATH: 101
  6    10.101.2.0/31   10.101.2.1    0            100         0       BE
      AS_PATH: 101
  7    10.102.10.0/24  10.101.2.1    0            100         0       BE
      AS_PATH: 101
  8    10.102.88.0/24  10.101.2.1    0            100         0       BE
      AS_PATH: 101
  9    10.128.1.0/31   10.101.2.1    0            100         0       BE
      AS_PATH: 101
 10    10.222.10.0/24   10.101.2.1    0            100         0       BE
      AS_PATH: 101
 11    10.25.222.2/31  10.101.2.1    0            100         0       BE
      AS_PATH: 101
 12    10.2.2.0/24     10.101.2.1    1            100         0       BE
      AS_PATH: 101
 13    10.122.125.0/24 10.101.2.1    1            100         0       BE
      AS_PATH: 101
 14    10.123.124.125/32 10.101.2.1    1            100         0       BE
      AS_PATH: 101

```

This example displays the details of advertised routes for VRF instance "red".

```

device# show ip bgp neighbors 10.5.5.5 advertised-routes vrf red
      There are 3 routes advertised to neighbor 10.5.5.5
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
      Prefix          Next Hop      MED          LocPrf      Weight  Status
  1    10.5.5.0/24     10.5.5.6      0            100         32768   BL
      AS_PATH:
  2    10.6.6.6/32     10.5.5.6      0            100         32768   BL
      AS_PATH:
  3    10.7.7.7/32     10.5.5.6      0            100         32768   BL
      AS_PATH:

```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp neighbors flap-statistics

Displays the route flap statistics for routes received from or sent to a BGP4 neighbor.

## Syntax

```
show ip bgp neighbors ip-addr flap-statistics [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example shows flap statistics for VRF instance "red".

```
device# show ip bgp neighbors 5.5.5.6 flap-statistics vrf red
Total number of flapping routes: 5
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp neighbors last-packet-with-error

Displays information about the last packet that contained an error from any of a device's neighbors.

## Syntax

```
show ip bgp neighbors ip-addr last-packet-with-error [ decode ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**decode**

Decodes last packet that contained an error from any of a device's neighbors.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ip bgp neighbors last-packet-with-error** command displays the following information.

Output field	Description
Total number of BGP Neighbors	The total number of configured neighbors for a device.
Last error	The error packet's contents decoded in a human-readable format or notification that no packets with an error were received.

# show ip bgp neighbors received

Displays Outbound Route Filters (ORFs) received from BGP4 neighbors of the device.

## Syntax

```
show ip bgp neighbors ip-addr received [ extended-community | prefix-filter ] [ rbridge-id { rbridge-id | all } ] vrf vrf-name ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**extended-community**

Displays the results for ORFs that use the BGP Extended Community Attribute.

**prefix-filter**

Displays the results for ORFs that are prefix-based.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

The following example displays prefix-based ORFs received for VRF instance "red".

```
device# show ip bgp neighbors 5.5.5.6 received prefix-filter vrf red
```

```
ip prefix-list: 3 entries
  seq 5 permit 5.5.5.5/32
  seq 10 permit 6.6.6.6/32
  seq 15 permit 100.1.1.0/24 ge 25 le 31
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp neighbors received-routes

Lists all route information received in route updates from BGP4 neighbors of the device since the soft-reconfiguration feature was enabled.

## Syntax

```
show ip bgp neighbors ip-addr received-routes [ detail ] [ rbridge-id { rbridge-id | all } ] vrf vrf-name ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**detail**

Displays detailed route information.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode



## Examples

This example displays the details of route updates for VRF instance "red".

```
device# show ip bgp neighbors 5.5.5.6 received-routes detail vrf red
  There are 3 received routes from neighbor 5.5.5.6
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
 1 Prefix: 5.5.5.0/24, Status: BI, Age: 0h0m36s
    NEXT_HOP: 5.5.5.6, Metric: 0, Learned from Peer: 5.5.5.6 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH:
      COMMUNITIES: 1:1 2:2 3:3 4:4
 2 Prefix: 6.6.6.6/32, Status: BI, Age: 0h0m36s
    NEXT_HOP: 5.5.5.6, Metric: 0, Learned from Peer: 5.5.5.6 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH:
      COMMUNITIES: 1:1 2:2 3:3 4:4
 3 Prefix: 7.7.7.7/32, Status: BI, Age: 0h0m36s
    NEXT_HOP: 5.5.5.6, Metric: 0, Learned from Peer: 5.5.5.6 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH:
      COMMUNITIES: 1:1 2:2 3:3 4:4
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp neighbors routes

Lists a variety of route information received in UPDATE messages from BGP4 neighbors.

## Syntax

```
show ip bgp neighbors ip-addr routes
```

```
show ip bgp neighbors ip-addr routes { best | not-installed-best | unreachable } [ rbridge-id { rbridge-id | all } ] vrf vrf-name
[ rbridge-id { rbridge-id | all } ]
```

```
show ip bgp neighbors ip-addr routes detail { best | not-installed-best | unreachable } [ rbridge-id { rbridge-id | all } ] vrf vrf-name
[ rbridge-id { rbridge-id | all } ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**best**

Displays routes received from the neighbor that are the best BGP4 routes to their destination.

**not-installed-best**

Displays routes received from the neighbor that are the best BGP4 routes to their destination but were not installed in the route table because the device received better routes from other sources.

**unreachable**

Displays routes that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf *vrf-name***

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example displays best-route information received in UPDATE messages for VRF instance "red":

```
device# show ip bgp neighbors 10.11.12.13 routes best vrf red
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp neighbors routes-summary

Lists all route information received in UPDATE messages from BGP4 neighbors.

## Syntax

```
show ip bgp neighbors ip-addr routes-summary [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example displays route summary information received in UPDATE messages for VRF instance "red":

```
device# show ip bgp neighbors routes-summary vrf red
  Total number of BGP Neighbors: 1
 1  IP Address: 5.5.5.6
Routes Accepted/Installed:3,  Filtered/Kept:0,  Filtered:0
  Routes Selected as BEST Routes:3
    BEST Routes not Installed in IP Forwarding Table:0
  Unreachable Routes (no IGP Route for NEXTHOP):0
  History Routes:0

NLRIs Received in Update Message:9,  Withdraws:0 (0),  Replacements:6
  NLRIs Discarded due to
    Maximum Prefix Limit:0,  AS Loop:0
    Invalid Nexthop:0,  Invalid Nexthop Address:0.0.0.0
    Invalid Confed aspath:0,  maxas-limit aspath:0
    Duplicated Originator_ID:0,  Cluster_ID:0

Routes Advertised:0,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:0,  Withdraws:0,  Replacements:0

Peer Out of Memory Count for:
  Receiving Update Messages:0,  Accepting Routes (NLRI):0
  Attributes:0,  Outbound Routes (RIB-out):0  Outbound Routes Holder:0
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp peer-group

Displays peer-group information.

## Syntax

```
show ip bgp peer-group peer-group-name [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*peer-group-name*

Specifies a peer group name.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Only the parameters that have values different from their defaults are listed.

## Examples

This example shows sample output from the **show ip bgp peer-group** command when no argument or keyword is used.

```
device# show ip bgp peer-group

1  BGP peer-group is vrfv4001_6, Remote AS: 34001
   Description: vrf_v4001_8192_131_v4001_6
   MD5 Password: $clVHfXIZUVFafVdae3x9WjhafXIZUVFafW0=
   Address family : IPV4 Unicast
   activate
   Address family : IPV6 Unicast
   activate
   Route Filter Policies:
   Route-map: (out) pass_vrf_v4001_6
   Currently there are no members.

6  BGP peer-group is VE_3001, Remote AS: 65530
   Description: 122_125_131_3001
   MD5 Password: $Wnx8fVp8T31aOFp9OFRWg==
   Address family : IPV4 Unicast
   activate
   Address family : IPV6 Unicast
   no activate
   Members:
   IP Address: 10.1.1.131
```

This example shows sample output from the **show ip bgp peer-group** command including information about dynamically created BGP neighbors.

```
device# show ip bgp peer-group

1  BGP peer-group is ebgp-peer, Remote AS: 800
   Description: ebgp-neighbors
   Multihop-EBGP: yes, ttl: default
   SoftInboundReconfiguration: yes
   Address family : IPV4 Unicast
   activate
   SendCommunity: yes
   SendCommunity extended: yes
   Route Filter Policies:
   Route-map: (out) test
   Address family : IPV6 Unicast
   no activate
   Address family : L2VPN EVPN
   no activate
   Currently there are no members.

2  BGP peer-group is ibgp-mh, Remote AS: 1234
   Alternate AS-Range: 700
   Multihop-EBGP: yes, ttl: default
   Updatesource: Loopback 5
   SoftInboundReconfiguration: yes
   Address family : IPV4 Unicast
   activate
   MaximumPrefixLimit: 100
   Address family : IPV6 Unicast
   no activate
   Address family : L2VPN EVPN
   activate
   SendCommunity extended: yes
   MaximumPrefixLimit: 10000
   Members:
   IP Address: 33.11.11.11
   IP Address: 33.44.44.44
```

```
show ip bgp peer-group
```

This example shows sample output from the **show ip bgp peer-group** command, including information about dynamically created BGP neighbors, when a peer group is specified.

```
device# show ip bgp peer-group ibgp-mh

1  BGP peer-group is ibgp-mh, Remote AS: 1234
   Alternate AS-Range: 700
   Multihop-EBGP: yes, ttl: default
   Updatesource: Loopback 5
   SoftInboundReconfiguration: yes
   Address family : IPV4 Unicast
   activate
   MaximumPrefixLimit: 100
   Address family : IPV6 Unicast
   no activate
   Address family : L2VPN EVPN
   activate
   SendCommunity extended: yes
   MaximumPrefixLimit: 10000
Members:
  IP Address: 33.11.11.11
  IP Address: 33.44.44.44
```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.
7.1.0	This command was modified to display information about dynamically created BGP neighbors.



# show ip bgp rbridge-id

Displays BGP4 route information by RBridge ID.

## Syntax

```
show ip bgp rbridge-id { rbridge-id | all } [ vrf vrf-name ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example displays route information for RBridge ID 5 and VRF instance "red".

```
device# show ip bgp rbridge-id 5 vrf red
```

## History

Release version	Command history
6.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ip bgp routes

Displays BGP4 route information that is filtered by the table entry at which the display starts.

## Syntax

```
show ip bgp routes [ num | ip-address/prefix | age num | as-path-access-list name | best | cidr-only | community-access-list
name | community-reg-expression expression | detail | local | neighbor ip-addr | nexthop ip-addr | no-best | not-
installed-best | prefix-list string | regular-expression name | route-map name | summary | unreachable ] [ rbridge-id
{ rbridge-id | all } ] | vrf vrf-name ]
```

## Parameters

*num*

Table entry at which the display starts.

*ip-address/prefix*

Table entry at which the display starts.

**age**

Displays BGP4 route information that is filtered by age.

**as-path-access-list**

Displays BGP4 route information that is filtered by autonomous system (AS)-path access control list (ACL).

**best**

Displays BGP4 route information that the device selected as best routes.

**cidr-only**

Displays BGP4 routes whose network masks do not match their class network length.

**community-access-list** *name*

Displays BGP4 route information for an AS-path community access list.

**community-reg-expression** *expression*

Displays BGP4 route information for an ordered community-list regular expression.

**detail**

Displays BGP4 detailed route information.

**local**

Displays BGP4 route information about selected local routes.

**neighbor** *ip-addr*

Displays BGP4 route information about selected BGP neighbors.

**nexthop** *ip-addr*

Displays BGP4 route information about routes that are received from the specified next hop.

**no-best**

Displays BGP4 route information that the device selected as not best routes.

**not-installed-best**

Displays BGP4 route information about best routes that are not installed.

**prefix-list** *string*

Displays BGP4 route information that is filtered by prefix list.

**regular-expression** *name*

Displays BGP4 route information about routes that are associated with the specified regular expression.

**route-map** *name*

Displays BGP4 route information about routes that use the specified route map.

**summary**

Displays BGP4 summary route information.

**unreachable**

Displays BGP4 route information about routes whose destinations are unreachable through any of the BGP4 paths in the BGP4 route table.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example shows sample input from the **show ip bgp routes** command when the **details** keyword is used.

```
device# show ip bgp routes detail
Total number of BGP Routes: 5
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
 1 Prefix: 10.2.2.0/24, Status: BE, Age: 12d15h19m24s
   NEXT_HOP: 10.101.2.1, Metric: 0, Learned from Peer: 10.101.2.1 (101)
   LOCAL_PREF: 100, MED: 1, ORIGIN: incomplete, Weight: 0
   AS_PATH: 101
   Adj_RIB_out count: 5, Admin distance 20
 2 Prefix: 10.11.11.1/32, Status: BE, Age: 12d15h19m24s
   NEXT_HOP: 10.101.2.1, Metric: 0, Learned from Peer: 10.101.2.1 (101)
   LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
   AS_PATH: 101
   Adj_RIB_out count: 5, Admin distance 20
 3 Prefix: 10.1.222.2/31, Status: BE, Age: 12d15h19m24s
   NEXT_HOP: 10.101.2.1, Metric: 0, Learned from Peer: 10.101.2.1 (101)
   LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
   AS_PATH: 101
   Adj_RIB_out count: 5, Admin distance 20
 4 Prefix: 10.122.125.0/24, Status: BE, Age: 12d15h19m24s
   NEXT_HOP: 10.101.2.1, Metric: 0, Learned from Peer: 10.101.2.1 (101)
   LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
   AS_PATH: 101
   Adj_RIB_out count: 5, Admin distance 20
 5 Prefix: 10.101.1.0/24, Status: BE, Age: 12d15h19m24s
   NEXT_HOP: 10.101.2.1, Metric: 0, Learned from Peer: 10.101.2.1 (101)
   LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
   AS_PATH: 101
   Adj_RIB_out count: 5, Admin distance 20
```

This example shows sample input from the **show ip bgp routes** command when the **summary** keyword is used.

```
device# show ip bgp routes summary
Total number of BGP routes (NLRIs) Installed      : 14
Distinct BGP destination networks                : 14
Filtered bgp routes for soft reconfig            : 0
Routes originated by this router                  : 0
Routes selected as BEST routes                    : 14
Routes Installed as BEST routes                   : 14
BEST routes not installed in IP forwarding table  : 0
Unreachable routes (no IGP route for NEXTHOP)   : 0
IBGP routes selected as best routes               : 0
EBGP routes selected as best routes               : 14
BEST routes not valid for IP forwarding table     : 0
```

This example shows sample input from the **show ip bgp routes** command when the **summary** and **vrf** keywords are used and a VRF is specified.

```
device# show ip bgp routes summary vrf green
Total number of BGP routes (NLRIs) Installed      : 4
Distinct BGP destination networks                : 4
Filtered bgp routes for soft reconfig            : 0
Routes originated by this router                  : 4
Routes selected as BEST routes                    : 4
Routes Installed as BEST routes                   : 4
BEST routes not installed in IP forwarding table  : 0
Unreachable routes (no IGP route for NEXTHOP)   : 0
IBGP routes selected as best routes               : 0
EBGP routes selected as best routes               : 0
BEST routes not valid for IP forwarding table     : 0
```

This example shows sample input from the **show ip bgp routes** command when the **vrf** keyword is used and a VRF is specified.

```
device# show ip bgp vrf green

Total number of BGP Routes: 4
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S
stale
Origin codes: i - IGP, e - EGP, ? - incomplete
  Network          Next Hop      RD           MED           LocPrf       Weight P
ath
*> 10.1.1.0/31     0.0.0.0      0             0             100          32768 ?
*> 10.3.1.0/31     0.0.0.0      0             0             100          32768 ?
*> 10.5.1.0/31     0.0.0.0      0             0             100          32768 ?
*> 10.6.1.254/31  0.0.0.0      0             0             100          32768 ?
```

This example shows sample input from the **show ip bgp routes** command when the **details** keyword is used.

Prefixes with path IDs for both incoming and outgoing paths are displayed.

Sender side display:

```
device# show ip bgp routes detail 1.0.0.0/24
Total number of BGP Routes: 6
Status A:AGGREGATE B:BEST b: NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: 1.0.0.0/24, Status: BME, Age: 0h17m10s
  NEXT_HOP: 20.1.0.1, Metric: 0, Learned from Peer: 20.1.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 0          Tx Path Id: 1
    Adj_RIB_out count: 10, Admin distance 20
2 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
  NEXT_HOP: 20.7.0.1, Metric: 0, Learned from Peer: 20.7.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 0          Tx Path Id: 2
3 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
  NEXT_HOP: 20.4.0.1, Metric: 0, Learned from Peer: 20.4.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 0          Tx Path Id: 3
4 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
  NEXT_HOP: 20.5.0.1, Metric: 0, Learned from Peer: 20.5.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 0          Tx Path Id: 4
5 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
  NEXT_HOP: 20.3.0.1, Metric: 0, Learned from Peer: 20.3.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 0          Tx Path Id: 5
6 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
  NEXT_HOP: 20.6.0.1, Metric: 0, Learned from Peer: 20.6.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 0          Tx Path Id: 6
7 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m12s
  NEXT_HOP: 20.2.0.1, Metric: 0, Learned from Peer: 20.2.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
Rx Path Id: 0          Tx Path Id: 0
```

Receiver side display:

```
device# show ip bgp routes detail 1.0.0.0/24
Total number of BGP Routes: 6
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: 1.0.0.0/24, Status: BME, Age: 0h17m10s
  NEXT_HOP: 20.1.0.1, Metric: 0, Learned from Peer: 20.1.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 6          Tx Path Id: 1
    Adj_RIB_out count: 10, Admin distance 20
2 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
  NEXT_HOP: 20.7.0.1, Metric: 0, Learned from Peer: 20.7.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 2          Tx Path Id: 2
3 Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
  NEXT_HOP: 20.4.0.1, Metric: 0, Learned from Peer: 20.4.0.1 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
  AS_PATH: 2
  Rx Path Id: 4          Tx Path Id: 3
```

```

4   Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
    NEXT_HOP: 20.5.0.1, Metric: 0, Learned from Peer: 20.5.0.1 (2)
    LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
    AS_PATH: 2
    Rx Path Id: 3           Tx Path Id: 4
5   Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
    NEXT_HOP: 20.3.0.1, Metric: 0, Learned from Peer: 20.3.0.1 (2)
    LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
    AS_PATH: 2
    Rx Path Id: 5           Tx Path Id: 5
6   Prefix: 1.0.0.0/24, Status: ME, Age: 0h17m10s
    NEXT_HOP: 20.6.0.1, Metric: 0, Learned from Peer: 20.6.0.1 (2)
    LOCAL_PREF: 100, MED: none, ORIGIN: igp, Weight: 0
    AS_PATH: 2
    Rx Path Id: 1           Tx Path Id: 6

```

## History

Release version	Command history
6.0.1	The <code>vrf vrf-name</code> parameter was added to support Multi-VRF.
7.0.0	Command output was modified to include details about BGP additional paths.

# show ip bgp routes community

Displays BGP4 route information that is filtered by community and other options.

## Syntax

```
show ip bgp routes community { num | aa:nn | internet | local-as | no-advertise | no-export } [ rbridge-id { rbridge-id | all } ] vrf  
vrf-name [ rbridge-id { rbridge-id | all } ]]
```

## Parameters

### community

Displays routes filtered by a variety of communities.

### *num*

Specifies a community number in the range from 1 to 4294967200.

### *aa:nn*

Specifies an autonomous system-community number.

### internet

Displays routes for the Internet community.

### local-as

Displays routes for a local sub-AS within the confederation.

### no-advertise

Displays routes with this community that cannot be advertised to any other BGP4 devices at all.

### no-export

Displays routes for the community of sub-ASs within a confederation.

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

### vrf *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode



## Examples

This example shows output from the **show ip bgp routes community** command when the **internet** keyword is used.

```
device# show ip bgp routes community internet
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop    MED      LocPrf    Weight    Status
1   10.2.2.0/24  10.101.2.1  1         100       0         BE
   AS_PATH: 101
2   10.11.11.1/32  10.101.2.1  0         100       0         BE
   AS_PATH: 101
3   10.1.222.2/31  10.101.2.1  0         100       0         BE
   AS_PATH: 101
4   10.122.125.0/24  10.101.2.1  0         100       0         BE
   AS_PATH: 101
5   10.101.1.0/24  10.101.2.1  0         100       0         BE
   AS_PATH: 101
6   10.11.122.0/24  10.101.2.1  0         100       0         BE
   AS_PATH: 101
7   10.101.2.0/31  10.101.2.1  0         100       0         BE
   AS_PATH: 101
8   10.102.10.0/24  10.101.2.1  0         100       0         BE
   AS_PATH: 101
9   10.102.88.0/24  10.101.2.1  0         100       0         BE
   AS_PATH: 101
10  10.128.1.0/31  10.101.2.1  0         100       0         BE
   AS_PATH: 101
11  10.222.10.0/24  10.101.2.1  0         100       0         BE
   AS_PATH: 101
12  10.25.222.2/31  10.101.2.1  0         100       0         BE
   AS_PATH: 101
13  10.122.125.0/24  10.101.2.1  1         100       0         BE
   AS_PATH: 101
14  10.123.124.125/32  10.101.2.1  1         100       0         BE
   AS_PATH: 101
```

## History

Release version	Command history
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ip bgp summary

Displays BGP information such as the local autonomous system number (ASN), maximum number of routes supported, and some BGP4 statistics.

## Syntax

```
show ip bgp summary [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

The following example displays summary BGP information.

```
device# show ip bgp summary

BGP4 Summary
Router ID: 10.122.122.122   Local AS Number: 122
Confederation Identifier: not configured
Confederation Peers:
Maximum Number of IP ECMP Paths Supported for Load Sharing: 8
Number of Neighbors Configured: 6, UP: 4
Number of Routes Installed: 14, Uses 1344 bytes
Number of Routes Advertising to All Neighbors: 70 (28 entries), Uses 1680 byte
s
Number of Attribute Entries Installed: 2, Uses 188 bytes
Neighbor Address  AS#      State   Time      Rt:Accepted  Filtered  Sent  ToSend
10.1.1.131       65530   CONN    12d13h14m  0            0         0    14
10.101.2.1       101     ESTAB   12d13h 5m  14          0         0    0
10.57.1.1        7000    ACTIV   12d13h14m  0            0         0    14
10.125.125.125   122     ESTAB   0h15m22s   0            0         14   0
10.154.154.154   122     ESTAB   12d13h 4m  0            0         14   0
10.155.155.155   122     ESTAB   12d13h 4m  0            0         14   0
```

The following example shows sample output from the **show ip bgp summary** command, including information about dynamically created BGP neighbors.

```
device# show ip bgp summary

BGP4 Summary
Router ID: 77.77.77.77   Local AS Number: 1234
Confederation Identifier: not configured
Confederation Peers:
Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
Number of Neighbors Configured: 5, UP: 3
Number of Routes Installed: 12, Uses 1440 bytes
Number of Routes Advertising to All Neighbors: 36 (12 entries), Uses 720 bytes
Number of Attribute Entries Installed: 2, Uses 230 bytes
*Dynamically created based on a listen range command
Dynamically created neighbors: 3/25(max)
Neighbor Address  AS#      State   Time      Rt:Accepted  Filtered  Sent    ToSend
*33.11.11.11     1234    ESTAB   1d 0h42m  11           0         1       0
*33.44.44.44     1234    ESTAB   1d 0h43m  0            0         12      0
*212.17.1.1      1234    OPENSp  0h 0m 0s  0            0         0       0
```

The following example shows sample output from the **show ip bgp summary** command including BGP automatic neighbor discovery configurations.

```
device# show ip bgp summary

BGP4 Summary
Router ID: 26.26.26.1   Local AS Number: 20
Confederation Identifier: not configured
Confederation Peers:
Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
Number of Neighbors Configured: 1, UP: 1
Number of Routes Installed: 0
Number of Routes Advertising to All Neighbors: 0 (0 entries)
Number of Attribute Entries Installed: 0
*Dynamically created based on a listen range command
Dynamically created neighbors: 0/100(max)
A: Auto Discovered Neighbors using LLDP
Auto Neighbors Count: 1
Neighbor Address  AS#      State   Time      Rt:Accepted  Filtered  Sent    ToSend
A23.23.23.3      30       ESTAB   19h 6m17s  0            0         0       0
```

## History

Release version	Command history
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.
7.1.0	This command was modified to display information about dynamically created BGP neighbors.
7.2.0	This command was modified to include BGP automatic neighbor discovery information.

# show ip community-list

Displays that status of BGP community lists.

## Syntax

```
show ip community-list [list_name [rbridge-id number] | rbridge-id list_name]
```

## Parameters

*list\_name*

Name of a BGP community list.

*rbridge-idnumber*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To view community list status for a specific list:

```
device# show ip community-list mycommunitylist
```

# show ip dhcp relay address interface

Displays IP DHCP relay addresses configured on supported interfaces.

## Syntax

**show ip dhcp relay address interface**

**show ip dhcp relay address interface** *<N>***gigabitethernet** *rbridge-id/slot/port*

**show ip dhcp relay address interface port-channel** *number* [**rbridge-id** { *rbridge-id* | *range* | **all** }]

**show ip dhcp relay address interface ve** *vlan-id* [**rbridge-id** { *rbridge-id* | *range* | **all** }]

## Parameters

**<N>****gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>****gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel interface. The range is from 1 through 6144.

**rbridge-id**

Specifies an RBridge, multiple RBridges, or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

*range*

Specifies multiple RBridge IDs. You can specify a range (for example, 3-5), a comma-separated list (for example, 1,3,5,6), or combine a range with a list (for example, 1-5,6,8). In a range string, no spaces are allowed.

**all**

Specifies all RBridges.

**ve** *vlan-id*

Specifies a virtual Ethernet (VE) interface.

**ethernet** *vlan-id*

Specifies a virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the **rbridge-id** parameter is omitted, IP DHCP relay addresses display for the local switch.

## Examples

The following example displays typical command output.

```
device# show ip dhcp relay address
DHCP Relay Agent Information Option Enabled
                                Rbridge Id:    2
                                -----
Interface                        Relay Address                VRF Name
-----
Te 2/2/1                          10.1.1.1                      Blue
Te 2/4/2                          20.1.1.1                      Blue
Te 2/5/4                          30.1.1.1                      Default-vrf
Ve 100                             40.1.1.1                      Green
```

The following example displays configured IPv4 DHCP relay addresses on a specified physical interface.

```
device# show ip dhcp relay address interface tengigabitethernet 2/0/37
DHCP Relay Information Option: Disabled
DHCP Remote ID (Type:Length:Vlan:Mac): 00:08:0000:0027f8cad501
                                Rbridge Id:    2
                                -----
Interface                        Relay Address                VRF Name
-----
Te 2/0/37                        8.8.8.8                      default-vrf
```

The following example displays configured IPv4 DHCP relay addresses on a specified port-channel and RBridge ID.

```
device# show ip dhcp relay address interface port-channel 10 rbridge-id 2
DHCP Relay Information Option: Disabled
DHCP Remote ID (Type:Length:Vlan:Mac): 00:08:0000:0027f8cad51d
                                Rbridge Id:    2
                                -----
Interface                        Relay Address                VRF Name
-----
Po 10                            20.1.1.1                    default-vrf
```

The following example displays configured IPv4 DHCP relay addresses on a specified VE interface and RBridge ID.

```
device# show ip dhcp relay address interface ve 10 rbridge-id 2
DHCP Relay Information Option: Disabled
DHCP Remote ID (Type:Length:Vlan:Mac): 00:08:000a:0027f8cad4d9
                                Rbridge Id:    2
                                -----
Interface                        Relay Address                VRF Name
-----
Ve 10                             4.4.4.4                      default-vrf
```

The following example displays configured IPv4 DHCP relay addresses on a specified VE interface and RBridges.

```
device# show ip dhcp relay address interface ve 10 rbridge-id 51,53
DHCP Relay Information Option: Disabled
DHCP Remote ID (Type:Length:Vlan:Mac): 00:08:000a:0027f852c94e
                                     Rbridge Id: 51
                                     -----
```

Interface	Relay Address	VRF Name
Ve 10	4.4.4.4	default-vrf

```
DHCP Relay Information Option: Disabled
DHCP Remote ID (Type:Length:Vlan:Mac): 00:08:000a:0027f8cb1377
                                     Rbridge Id: 53
                                     -----
```

Interface	Relay Address	VRF Name
Ve 10	14.1.1.1	default-vrf

## History

Release version	Command history
5.0.1	This command was introduced.
6.0.2a	This command was modified to support DHCP Relay Information Option.
7.0.0	This command was modified to support port-channels.

## Related Commands

[ip dhcp relay address](#), [show ip dhcp relay address rbridge-id](#)

# show ip dhcp relay address rbridge-id

Displays IP DHCP relay addresses.

## Syntax

```
show ip dhcp relay address rbridge-id rbridge-id | all | range
```

## Command Default

If the *rbridge-id* parameter is omitted, IP DHCP relay addresses display for the local switch.

## Parameters

*rbridge-id*

Specifies an RBridge ID. You can specify multiple RBridge IDs, separated by commas.

all

Specifies all RBridge IDs.

range

A range of RBridge IDs separated by dashes or commas.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays the IP address and Virtual Routing and Forwarding (VRF) name for all interfaces with configured IP DHCP relay addresses on a local switch, specific switches, or all switches in a VCS Fabric. No spaces are allowed in the *range* string. The range does not need to be contiguous (for example, 1-2,5).

A range of RBridge IDs can be separated by dashes or commas as follows:

1-3 - RBridge ID 1 through 3 1-3, 5 - RBridge ID 1 through 3 and RBridge ID 5 1, 3, 5, 6 - RBridge ID 1, 3, 5, and 6

## Examples

The following example displays addresses configured on a specific RBridge ID.

```
device# show ip dhcp relay address rbridge-id 2
DHCP Relay Information Option: Disabled
                               Rbridge Id:    2
                               -----
Interface      Relay Address      VRF Name
-----
Te 2/2/1      10.1.1.1              Blue
Te 2/4/2      20.1.1.1              Blue
Te 2/5/4      30.1.1.1              Default-vrf
Te 2/6/6      40.1.1.1              Green
```



The following example displays addresses configured on all switches in a virtual fabric cluster.

```
device# show ip dhcp relay address rbridge-id all

DHCP Relay Information Option: Disabled
                                Rbridge Id:    1
                                -----
Interface          Relay Address          VRF Name
-----
Te 1/0/24          2.3.4.5                default-vrf
Ve 300             10.0.1.2               default-vrf

DHCP Relay Information Option: Disabled
                                Rbridge Id:    3
                                -----
Interface          Relay Address          VRF Name
-----
Ve 300             10.0.0.5               default-vrf
```

## History

Release version	Command history
7.0.0	This command was modified to display the DHCP Relay Agent information Option (Option-82) in the output.

## Related Commands

[ip dhcp relay address](#), [show ip dhcp relay address interface](#)

# show ip dhcp relay gateway

Displays IP DHCP Relay gateway addresses.

## Syntax

```
show ip dhcp relay gateway [rbridge-id rbridge-id | all | range]
```

## Command Default

If the `rbridge-id` parameter is omitted, the gateway addresses display for the local switch.

## Parameters

*rbridge-id*

Specific RBridge identification. You can specify multiple RBridge IDs, separated by commas.

**all**

Specifies all RBridges.

*range*

A range of RBridge IDs separated by a dashes or commas, for example:

- 1-3 - RBridge ID 1 through 3,
- 1-3, 5 - RBridge ID 1 through 3 and RBridge ID 5
- 1, 3, 5, 6 - RBridge ID 1, 3, 5, and 6

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the IP address and Virtual Routing and Forwarding (VRF) name for all interfaces with configured IP DHCP Relay gateway addresses on a local switch, specific switches, or all switches in a VCS Fabric. No spaces are allowed in the range string. The range does not need to be contiguous (for example, 1-2,5).

## Examples

To display addresses configured on a specific RBridge ID:

```
device# show ip dhcp relay gateway address rbridge-id 2
```

```

                                Rbridge Id:    2
                                -----
DHCP Relay Information Option: Enabled
Interface      -----      Relay Address      VRF Name
-----
Te 2/2/1      10.1.1.1          Blue
Te 2/4/2          20.1.1.1          Blue
Te 2/5/4          30.1.1.1          Default-vrf
Te 2/6/6          40.1.1.1          Green

```

To display addresses configured on all switches in a cluster:

```
device# show ip dhcp relay gateway address rbridge-id all
```

```

                                Rbridge Id:    1
                                -----
DHCP Relay Information Option: Enabled
Interface                       Relay Address                       VRF Name
-----                       -
Te 1/0/24                       2.3.4.5                             default-vrf
Ve 300                           10.0.1.2                             default-vrf

                                Rbridge Id:    3
                                -----
DHCP Relay Information Option: Enabled
Interface                       Relay Address                       VRF Name
-----                       -
Ve 300                           10.0.0.5                             default-vrf

```

## History

Release version	Command history
7.0.1	Output updated
4.1.3	This command was introduced.

# show ip dhcp relay statistics

Displays the general information about the DHCP Relay function.

## Syntax

```
show ip dhcp relay statistics [ ip-address ip-addr ] [ rbridge-id rbridge-id | all | range ]
```

## Command Default

If the **rbridge-id** parameter is omitted, IP DHCP Relay statistics display for the local switch. If the **ip-address** parameter is omitted, statistics display for all configured addresses on defined switches.

## Parameters

**ip-address** *ip-addr*

IPv4 address of DHCP server where client requests are to be forwarded.

**rbridge-id** *rbridge-id*

Specifies an RBridge. You can specify multiple RBridge IDs, separated by commas.

**all**

Specifies all RBridges.

*range*

A range of RBridge IDs separated by a dashes or commas, for example:

1-3 - RBridge ID 1 through 3  
 1-3, 5 - RBridge ID 1 through 3 and RBridge ID 5  
 1, 3, 5, 6 - RBridge ID 1, 3, 5, and 6

## Modes

Privileged EXEC mode

## Usage Guidelines

No spaces are allowed in the *range* string. The range does not need to be contiguous (for example, 1-2,5).

You can also specify **all** for all RBridge IDs in a cluster. To display addresses for configured interfaces on a local switch, an RBridge ID parameter is not required.

The **show ip dhcp relay statistics** command displays the following information about the IP DHCP Relay function for IP DHCP Relay addresses configured on a local switch, specific switches, or all switches in a cluster:

The **show ip dhcp relay statistics** command displays the following information about the IP DHCP Relay function for IP DHCP Relay addresses configured on the switch:

- DHCP Server IP Address configured in the switch.
- Number of DHCP DISCOVERY, OFFER, REQUEST, ACK, NAK, DECLINE, and RELEASE packets received.
- Number of DHCP client packets received (on port 67) and relayed by the Relay Agent.

- Number of DHCP server packets received (on port 67) and relayed by the Relay Agent.

## Examples

To display statistics for a local switch:

```
device# show ip dhcp relay statistics
DHCP Relay Statistics - Rbridge Id: 3
-----
Address      Disc. Offer      Req.      Ack      Nak      Decline
Release      Inform
-----
-----
10.1.0.1      400      100      2972      2968
0            0
20.2.0.1      400      100      2979      2975      0
0            0
30.3.0.1      400      100      3003      2998      0
0            0
40.4.0.1      400      100      3026      3018      0
0            0
Active Clients: 400
Clients to Restore: 0
Client Packets: 12780
Server Packets: 12359
Timed Out: 0
No Offers: 0
```

To display statistics for specific RBridge IDs:

```
device# show ip dhcp relay statistics rbridge-id 1,3
DHCP Relay Statistics - Rbridge Id: 1
-----
Address      Disc. Offer      Req.      Ack      Nak      Decline
Release      Inform
-----
-----
2.3.4.5      300      100      1211      2968      0
0            0
10.0.1.2      300      100      1207      2975      0
0            0
Client Packets: 2701
Server Packets: 2932

DHCP Relay Statistics - Rbridge Id: 3
-----
Address      Disc. Offer      Req.      Ack      Nak      Decline
Release      Inform
-----
-----
10.0.0.5      0      0      0      0
0            0
10.0.1.2      0      0      0      0
0            0
Client Packets: 0
Server Packets: 0
```

## Related Commands

[show ip dhcp relay address interface](#), [show ip dhcp relay address rbridge-id](#)

# show ip dns

Displays Domain Name System (DNS) status.

## Syntax

```
show ip dns
```

## Modes

Privileged EXEC mode

## Examples

To view DNS status:

```
device# show ip dns
search domain1
nameserver 1.1.1.1
```

# show ip extcommunity-list

Displays that status of BGP extended community lists.

## Syntax

```
show ip extcommunity-list [list_name [rbridge-id number] | rbridge-id list_name]
```

## Parameters

*list\_name*

Name of a BGP extended community list.

*rbridge-idnumber*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To view extended community list status for a specific list:

```
device# show ip extcommunity-list myextcommunitylist
```

# show ip fabric-virtual-gateway

Displays IPv4 Fabric-Virtual-Gateway session details.

## Syntax

```
show ip fabric-virtual-gateway { detail | summary | interface ve vlan-id } [ rbridge-id { rbridge-id | all }
```

## Parameters

### detail

Lists the IP Fabric-Virtual-Gateway configuration in detail.

### summary

Lists a summary of the IP Fabric-Virtual-Gateway configuration.

### interface ve *vlan-id*

Displays IPv4 Fabric-Virtual-Gateway configuration for the specified VE interface.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Examples

The following example displays the results of the **show ip fabric virtual gateway** command.

```
device# show ip fabric-virtual-gateway
=====Rbridge-id:3=====
Total number of IPv4 Fabric Virtual Gateway sessions   : 3
Total number of sessions in Active state             : 0
Total number of sessions in InActive state           : 0
Total number of sessions in Init state               : 3
Gateway MAC address: 02e0.5200.01ff
Configuration disabled due to reason(s): Gateway MAC address conflict
```

Interface	Admin State	State	Gateway IP Address	ARP Responder	Load Balancing	Threshold Priority	Track Priority
Ve 100	Enabled	Init	8.3.1.100/24		Enabled	unset	0
Ve 200	Enabled	Init	8.3.2.100/24		Enabled	unset	0
Ve 300	Enabled	Init	8.3.3.100/24		Enabled	unset	0



The following example displays IPv4 Fabric-Virtual-Gateway session details.

```
device# show ip fabric-virtual-gateway detail

=====Rbridge-id:1=====
Total number of IPv4 Fabric Virtual Gateway sessions   : 9
Total number of sessions in Active state             : 9
Total number of sessions in InActive state           : 0
Total number of sessions in Init state                : 0

Interface: Ve 100; Ifindex: 1207959652
  Admin Status: Enabled
  Description :
  Address family: IPV4      State: Active
  ARP responder Rbridge-id: 2
  Gateway IP: 10.1.1.101/24
  Gateway MAC Address: 02e0.5200.01ff
  Load balancing configuration: Enabled
  Load balancing current status: Enabled
  Load balancing threshold priority: unset
  Gratuitous ARP Timer: 10 sec
  Hold time: 0 sec (default: 0 sec)
  Total no. of state changes: 1
  Gratuitous ARP Sent: 0
  Last state change: 0d.0h.37m.41s ago
  Track Priority: 0
```

The following example displays IPv4 Fabric-Virtual-Gateway session details for a specified VE interface.

```
device# show ip fabric-virtual-gateway int ve 100

=====Rbridge-id:1=====
Interface  Admin   State      Gateway      ARP      Load      Threshold  Track
          State      State      IP Address   Responder Balancing Priority    Priority
-----
Ve 100    Enabled Active     10.1.1.101/24 Rbr-id 2 Enabled   unset      0
sw0#
```

## History

Release version	Command history
5.0.1	This command was introduced.
6.0.1	The show outputs were updated.

# show ip igmp groups

Displays information related to learned groups in the IGMP protocol module.

## Syntax

```
show ip igmp groups [ rbridge-id { rbridge-id | all } ]
```

```
show ip igmp groups [ A.B.C.D ] [ detail ] [ rbridge-id { rbridge-id | all } ]
```

```
show ip igmp groups interface { <N>gigabitethernet rbridge-id / slot / port | port-channel number | tunnel number | vlan vlan_id } [ A.B.C.D ] [ detail ]
```

```
show ip igmp groups interface ve vlan_id [ A.B.C.D ] [ detail ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

all

Specifies all RBridges.

*A.B.C.D*

Specifies the group address, as a subnet number in dotted decimal format (for example, 225.0.0.10), as the allowable range of addresses included in the multicast group.

detail

Displays detailed information.

interface

Specifies an interface type.

<N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

tunnel *tunnel-id*

Specifies a tunnel interface. The range is from 1 through 100000.

**vlan** *vlan\_id*  
Specifies a VLAN interface.

**ve** *vlan\_id*  
Specifies a virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the IGMP database, including configured entries for either all groups on all interfaces, or all groups on specific interfaces, or specific groups on specific interfaces.

The remote RBridge information is not displayed when the detail and interface operands are used.

When **rbridge-id** is specified, groups learned on Layer 3 interfaces for that particular RBridge ID are displayed. However, groups learned on Layer 2 interfaces from all the nodes in an entire cluster are displayed.

If **rbridge-id** is not specified, IGMP groups on Layer 3 interfaces of the node on which the command is executed are displayed. Groups learned on Layer 2 interfaces from all the nodes in the cluster are displayed.

When **rbridge-id all** is specified, all groups from all the nodes in the cluster are displayed.

## Examples

The following example is a detailed output.

```
device# show ip igmp groups interface vlan 101 detail
Interface:      Te 19/2/2
Group:          225.0.0.1
Uptime:         00:03:54
Expires:        00:00:35
Last reporter:  192.85.1.5
Last reporter mode: IGMP V2

Interface:      Te 19/2/1
Group:          225.0.0.1
Uptime:         00:04:16
Expires:        00:00:07
Last reporter:  192.85.1.6
Last reporter mode: IGMP V2
```

The following example is a VE output.

```
device# show ip igmp groups interface ve 2006
Total Number of Groups: 1
IGMP Connected Group Membership
Group Address  Interface  Uptime      Expires     Last Reporter
226.226.1.1    Vlan 2006  00:00:09    00:04:03   112.26.1.25
Member Ports: Te 125/2/12
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

# show ip igmp interface

Displays Layer 3 IGMP interface configuration information.

## Syntax

```
show ip igmp interface [ rbridge-id { rbridge-id | range | all } ]
```

```
show ip igmp interface [ <N>gigabitethernet rbridge-id / slot / port | port-channel number | vlan vlan_id }
```

```
show ip igmp interface ve vlan_id [ rbridge-id { rbridge-id | range | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

### ve *vlan\_id*

Specifies a Virtual Ethernet (VE) interface.

### vlan *vlan\_id*

Specifies a VLAN interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

When the **rbridge-id** option is specified, details for the VE interface on that particular RBridge are displayed. If **rbridge-id** is not specified, details for the VE interface on the node on which the command is executed is displayed. When **rbridge-id all** is specified, all VE interfaces with that RBridge from all the nodes in the cluster are displayed.

## Examples

The following example displays IGMP protocol information for VLAN 1.

```
device# show ip igmp interface vlan 1

Interface Vlan 1
  IGMP Snooping disabled
  IGMP Snooping fast-leave disabled
  IGMP Snooping querier disabled
  Number of router-ports: 0
```

The following example displays IGMP protocol information for RBridge ID 2.

```
device# show ip igmp interface rbridge-id 2

RbridgeId: 2
interface Ve 20
  IGMP enabled
  IGMP query interval 125 seconds
  IGMP other-querier interval 255 seconds
  IGMP query response time 10 seconds
  IGMP last-member query interval 1 seconds
  IGMP startup-query interval 31 seconds
  IGMP startup-query count 2
  IGMP last-member query count 2
  IGMP robustness variable 2
  IGMP immediate-leave disabled
  IGMP querier 20.1.1.1(this system)
  IGMP version 2
```

The following example displays IGMP protocol information for interface Te 2/0/47.

```
device# show ip igmp interface te 2/0/47

  IGMP enabled
  IGMP query interval 125 seconds
  IGMP other-querier interval 255 seconds
  IGMP query response time 10 seconds
  IGMP last-member query interval 1 seconds
  IGMP startup-query interval 31 seconds
  IGMP startup-query count 2
  IGMP last-member query count 2
  IGMP robustness variable 2
  IGMP immediate-leave disabled
  IGMP querier 47.1.1.1(this system)
  IGMP version 2
```

The following example displays IGMP protocol information for port-channel 1.

```
device# show ip igmp interface port-channel 1

Interface pol
  IGMP disabled
```

# show ip igmp snooping

Displays IGMP snooping information.

## Syntax

```
show ip igmp snooping [ interface vlan vlan_id | mrouter interface vlan vlan_id ]
```

## Parameters

**interface vlan** *vlan\_id*

Specifies which VLAN interface to display the snooping configuration related information.

**mrouter interface vlan** *vlan\_id*

Specifies which VLAN interface to display the snooping configuration related information.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **show ip igmp snooping** command to display IGMP snooping information, display multicast router port related information for the specified VLAN, or to display snooping statistics for the specified VLAN in the IGMP protocol module.

## Examples

To display IGMP snooping information for VLAN 5:

```
device# show ip igmp snooping interface vlan 5
```

# show ip igmp static-groups

Displays information about the Internet Group Management Protocol (IGMP) static groups.

## Syntax

```
show ip igmp static-groups { A.B.C.D | detail rbridge-id | interface [ <N>gigabitethernet | port-channel interface number |
tunnel tunnel ID | ve interface number | vlan interface number ] | rbridge-ID rbridge-id }
```

## Parameters

*A.B.C.D*

Specifies the group address, as a subnet number in dotted decimal format (for example, 225.0.0.10), as the allowable range of addresses included in the multicast group.

**detail**

Displays detailed information about the IP IGMP static groups.

**rbridge-ID**

Specifies the RBridge ID.

**interface**

Specifies the interface type, which can be <*N*>gigabitethernet, port-channel, tunnel, ve, vlan, or rbridge-id.

## Modes

Privileged EXEC mode

## Command Output

The **show ip igmp static-groups** command displays the following information:

Output field	Description
Group address	The IP address of the IGMP group.
Interface	The name of the interface.
Uptime	The amount of time the group membership has been up.
Expires	The time by which the group membership expires.
Last Reporter	The most recent source that has joined the multicast group.

## Examples

The following example displays details for VLAN 10.

```
device# show ip igmp static-groups interface vlan 10 detail
Interface:      Te 2/0/4
Group:          230.1.1.1
Uptime:        17:05:39
Expires:       Never
Last reporter:  Static
Last reporter mode: IGMP V2

Interface:      Te 2/0/4
Group:          230.1.1.2
Uptime:        17:05:39
Expires:       Never
Last reporter:  Static
Last reporter mode: IGMP V2

Interface:      Po 20
Group:          225.1.1.1
Uptime:        17:05:25
Expires:       Never
Last reporter:  Static
Last reporter mode: IGMP V2

Interface:      Po 20
Group:          225.1.1.2
Uptime:        17:05:25
Expires:       Never
Last reporter:  Static
Last reporter mode: IGMP V2
```

The following example displays information about the IP IGMP static groups.

```
device# show ip igmp static-groups

Total Number of Groups: 4
IGMP Connected Group Membership
Group Address      Interface      Uptime      Expires      Last Reporter
230.1.1.1          Vlan 10       17:04:49    Never        Static
  Member Ports: Te 2/0/4
230.1.1.2          Vlan 10       17:04:49    Never        Static
  Member Ports: Te 2/0/4
225.1.1.1          Po 20         17:04:35    Never        Static
225.1.1.2          Po 20         17:04:35    Never        Static
```

The following example displays information about the IP IGMP static groups for VLAN 10.

```
device# show ip igmp static-groups interface vlan 10

Total Number of Groups: 2
IGMP Connected Group Membership
Group Address      Interface      Uptime      Expires      Last Reporter
230.1.1.1          Vlan 10       17:05:09    Never        Static
  Member Ports: Te 2/0/4
230.1.1.2          Vlan 10       17:05:09    Never        Static
  Member Ports: Te 2/0/4
```

The following example displays information for port channel 20.

```
device# show ip igmp static-groups interface po 20

Total Number of Groups: 2
IGMP Connected Group Membership
Group Address      Interface      Uptime      Expires      Last Reporter
225.1.1.1          Po 20         17:05:04    Never        Static
225.1.1.2          Po 20         17:05:04    Never        Static
```



The following example displays details about the IGMP static groups for multicast group IP address 230.1.1.1.

```
device# show ip igmp static-groups 230.1.1.1
```

```
Total Number of Groups: 1
IGMP Connected Group Membership
Group Address      Interface      Uptime        Expires      Last Reporter
230.1.1.1         Vlan 10       17:07:42      Never        Static
Member Ports: Te 2/0/4
```

The following example displays details about the IGMP static groups for multicast group IP address on RBridge-ID 2.

```
device# show ip igmp static-groups 230.1.1.1 rbridge-id 2
```

```
Total Number of Groups: 1
IGMP Connected Group Membership
Group Address      Interface      Uptime        Expires      Last Reporter
230.1.1.1         Vlan 10       17:07:58      Never        Static
Member Ports: Te 2/0/4
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show ip igmp statistics interface

Displays IGMP statistics for an interface.

## Syntax

```
show ip igmp statistics interface { <N>gigabitethernet rbridge-id / slot / port | port-channel number | vlan vlan_id }
show ip igmp statistics interface ve vlan-id [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### rbridge-id

Specifies an RBridge ID.

#### slot

Specifies a valid slot number.

#### port

Specifies a valid port number.

### port-channel number

Specifies a port-channel interface. The range is from 1 through 6144.

### vlan vlan\_id

Specifies which VLAN interface to display the snooping configuration related information.

### ve vlan\_id

Specifies a virtual Ethernet (VE) interface.

### rbridge-id

Specifies an RBridge or all RBridges.

#### rbridge-id

Specifies an RBridge ID.

#### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

If a VE is specified:

- If an RBridge ID is not specified, details for the VE interface on the node on which the command is executed are displayed.

- When an RBridge ID is specified, details for the VE interface on that particular RBridge are displayed.
- When **rbridge-id all** is specified, all the nodes in the cluster on that VE are displayed.

## Examples

The following example displays the output of the **show ip igmp statistics interface** command.

```
device# show ip igmp statistics interface vlan 1

IGMP packet statistics for all interfaces in Vlan 1:
IGMP Message type      Edge-Received   Edge-Sent   Edge-Rx-Errors   ISL Received
Membership Query       0               0           0                 0
V1 Membership Report   0               0           0                 0
V2 Membership Report   0               0           0                 0
Group Leave            0               0           0                 0
V3 Membership Report   0               0           0                 0
PIM hello              0               0           0                 0

IGMP Error Statistics:
Unknown types          0
Bad Length             0
Bad Checksum           0
```

# show ip interface

Displays the IP interface status and configuration of all interfaces or a specified interface.

## Syntax

```
show ip interface [ brief [ rbridge-id { rbridge-id | all } ] | <N>gigabitethernet rbridge-id/slot/port | loopback number | port-  
channel number | ve vlan_id ]
```

## Parameters

### brief

Specifies to display a brief summary of IP interface status and configuration.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### loopback *number*

Specifies to display the loopback interface number. Valid values range from 1 through 255.

### port-channel *number*

Specifies to display the port-channel number. Valid values range from 1 through 6144.

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface (VLAN interface number).

## Modes

Privileged EXEC mode

## Usage Guidelines

Note the following with respect to the **show ip interface brief** command:

- The command **show ip interface brief rbridge-id** *rbridge-id* provides information about all physical, loopback, and switched virtual interfaces (SVIs) specific to the given *rbridge-id*.
- The **show ip interface brief rbridge-id all** command provides information about all physical, loopback, and SVIs for all nodes in a cluster.
- If the **rbridge-id** option is not used, information about physical, loopback, and SVIs is shown for the local node only.

Note the following with respect to the **show ip interface loopback** command:

- The command **show ip interface loopback rbridge-id** *rbridge-id* shows the details of loopback interfaces for the given *rbridge-id*.
- The command **show ip interface loopback rbridge-id all** shows the details of loopback interfaces for all nodes in a cluster.
- If the **rbridge-id** option is not used, information about loopback interfaces is shown for the local node only.

Note the following with respect to the **show ip interface ve** command:

- The command **show ip interface ve rbridge-id** *rbridge-id* provides information about SVIs specific to the given *rbridge-id*.
- The command **show ip interface ve rbridge-id all** provides information about SVIs for all nodes in a cluster.

If the **rbridge-id** option is not used, information about SVIs is shown for the local node only on the Extreme VDX family of switches.

The **show ip interface** command displays information for numbered and unnumbered IP interfaces.

Reason codes, their causes, and recommended actions are provided to diagnose and resolve the causes of "Protocol" = "down". Reason codes are displayed only on platforms with optics support. RJ-45 ports and chassis blades do not support reason codes. Refer to the following table:

**TABLE 17** Reason codes, causes, and actions to be taken

Reason code	Cause	Action to be taken
SFP Absent	SFP is not inserted into the port.	Insert a valid SFP.
Signal not detected	Cable is not connected to the inserted SFP, either on the local or remote side or both. This error is also seen if the peer port is in the "administratively down" state.	Connect a cable between both the remote and local side. Enable the port on the peer side.
Signal fault detected	If a port observes any local or remote fault, this reason code is displayed. The problem is considered to be on the line side, which could be caused by either the local or the remote port.	Execute the <b>shutdown</b> command, followed by the <b>no shutdown</b> command, on the ports at both ends of the cable. If the issue persists, remove and reinsert the cable and module, then repeat the above.  If the issue persists, replace the SFP and repeat all of the above.
Signal sync is not detected	If either side of the link fails to attain synchronization, this reason code is displayed. The problem could be with either the local or the remote side, or both	Execute the <b>shutdown</b> command, followed by the <b>no shutdown</b> command, on the ports at both ends of the cable.

## Examples

The following example displays summary information for all interfaces.

```
device# show ip interface brief
```

Interface	IP-Address	Status	Protocol
Port-channel 10	unassigned	up	down
Port-channel 11	unassigned	up	down
Port-channel 12	unassigned	up	down
Port-channel 13	unassigned	up	up
Port-channel 14	unassigned	up	down
Port-channel 15	unassigned	up	up
Ten Gigabit Ethernet 1/0/0	unassigned	up	up
Ten Gigabit Ethernet 1/0/1	unassigned	up	down (SFP Absent)
Ten Gigabit Ethernet 1/0/2	unassigned	up	up
Ten Gigabit Ethernet 1/0/3	unassigned	up	up
Ten Gigabit Ethernet 1/0/4	unassigned	up	down (ISL) (Signal sync is not detected)
Ten Gigabit Ethernet 1/0/5	unassigned	up	down (Signal not detected)
Ten Gigabit Ethernet 1/0/6	unassigned	up	down (Signal not detected)
Ten Gigabit Ethernet 1/0/7	unassigned	up	up
Ten Gigabit Ethernet 1/0/8	unassigned	up	up
Ten Gigabit Ethernet 1/0/9	unassigned	up	up
Ten Gigabit Ethernet 1/0/10	unassigned	up	down (Signal not detected)
Ten Gigabit Ethernet 1/0/11	unassigned	up	down (SFP Absent)
Ten Gigabit Ethernet 1/0/12	unassigned	up	up
Ten Gigabit Ethernet 1/0/13	unassigned	up	up
Ten Gigabit Ethernet 1/0/14	unassigned	up	down (SFP Absent)
Ten Gigabit Ethernet 1/0/15	unassigned	up	up
Ten Gigabit Ethernet 1/0/16	unassigned	up	down (Signal not detected)
Ten Gigabit Ethernet 1/0/17	unassigned	up	up
Ten Gigabit Ethernet 1/0/18	unassigned	up	down (Signal not detected)
Ten Gigabit Ethernet 1/0/19	unassigned	up	up
Ten Gigabit Ethernet 1/0/20	unassigned	up	up
Ten Gigabit Ethernet 1/0/21	unassigned	up	up
Ten Gigabit Ethernet 1/0/22	unassigned	up	up
Ten Gigabit Ethernet 1/0/23	unassigned	up	up
Vlan 1	unassigned	administratively down	down
Vlan 100	unassigned	administratively down	down
Vlan 200	unassigned	administratively down	down

The following example displays port-security status when the port-security feature is applied.

```
device# show ip interface brief
```

Interface	IP-Address	Status	Protocol
Port-channel 1	unassigned	up	up
TenGigabitEthernet 0/1	unassigned	up	up
TenGigabitEthernet 0/2	unassigned	admin-down	down "Port security violation"
TenGigabitEthernet 0/3	unassigned	admin-down	down
TenGigabitEthernet 0/4	unassigned	admin-down	down "Port security violation"

The following example displays the IP interface status of a 1-gigabit Ethernet port.

```
device# show ip interface gigabitethernet 1/0/1
```

```
Gigabit Ethernet 1/0/1 is up protocol is up
IP unassigned
IP MTU is 0
Proxy Arp is not Enabled
ICMP unreachable are always sent
ICMP mask replies are never sent
IP fast switching is enabled
RPF mode is Loose
VRF : default-vrf
```

The following example displays the IP interface status of an unnumbered 10-gigabit Ethernet port.

```
device# show ip interface tengigabitethernet 32/0/7:1
TenGigabitEthernet 32/0/7:1 is up protocol is up
Primary Internet Address is 32.32.32.32/32 broadcast is 30.0.0.30
Donor Interface: Loopback 2
IP Unnumbered is enabled
IP MTU is 1500
Proxy Arp is Enabled
IP fast switching is enabled
RPF mode is Strict
Vrf : default-vrf
```

The following example displays summary information, including down status resulting from repeated MAC-move detection.

```
device# show ip interface brief
Interface          IP-Address      Status Protocol
=====
Port-channel 1    unassigned      up    up
TenGigabitEthernet 0/1    unassigned      up    up
TenGigabitEthernet 0/2    unassigned      admin-down down "Repeated MAC-move Detection"
TenGigabitEthernet 0/3    unassigned      admin-down down
TenGigabitEthernet 0/4    unassigned      admin-down down
```

The following example displays summary information, including IP unnumbered interfaces.

```
device# show ip interface brief
Flags: U - Unnumbered interface
Interface          IP-Address      Vrf              Status
Protocol
=====
FortyGigabitEthernet 1/0/49    unassigned      default-vrf      up                down
FortyGigabitEthernet 1/0/50    unassigned      default-vrf      up                down
FortyGigabitEthernet 1/0/51    unassigned      default-vrf      up                down
FortyGigabitEthernet 1/0/52    unassigned      default-vrf      up                down
TenGigabitEthernet 1/0/1     1.1.1.1        default-vrf      up                down
TenGigabitEthernet 1/0/2     1.1.1.1 (U)    default-vrf      up                down
TenGigabitEthernet 1/0/3     unassigned      default-vrf      up                down
```

## History

Release version	Command history
7.0.0	This command was modified for unnumbered IP interfaces.
7.1.0	This command was modified to include reason codes for protocol down.
7.2.0	This command was modified for display of (unicast) Reverse Path Forwarding (RPF) mode.

## Related Commands

[show interface](#), [show mac-address-table mac-move](#)

# show ip interface loopback

Displays loopback information.

## Syntax

```
show ip interface loopback id [ rbridge-id { rbridge-id | all } ]
```

## Parameters

- id**  
Displays the information for the designated loopback.
- rbridge-id**  
Specifies an RBridge or all RBridges.
  - rbridge-id*  
Specifies an RBridge ID.
- all**  
Specifies all RBridges.

## Modes

Privileged EXEC mode

## Related Commands

[show ip interface](#), [show ip interface ve](#)



# show ip interface ve

Displays virtual Ethernet (VE) port information.

## Syntax

```
show ip interface ve id [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*id*

Displays the information for the designated loopback.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Related Commands

[show ip interface](#), [show ip interface loopback](#)

# show ip next-hop

Displays forwarding information related to the next-hop ID.

## Syntax

```
show ip next-hop { slot line_card_number } [ A.B.C.D | A.B.C.D/M ] vrf vrf-name [ A.B.C.D | A.B.C.D/M ]
```

## Parameters

**slot** *line\_card\_number*  
Specifies a slot.

*A.B.C.D/M*  
Specifies an IPv4 address with optional mask.

**vrf** *vrf-name*  
Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

To view forwarding information related to the next-hop ID for VRF "myvrf":

```
device# show ip next-hop vrf myvrf
```

# show ip ospf

Displays OSPF information.

## Syntax

```
show ip ospf [ vrf name [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

### **vrf *name***

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

### **rbridge-id**

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

When the RBridge ID is not specified, the output from the local node is displayed.

When the RBridge ID is specified, data from the corresponding specified RBridge is displayed.

When **all** is specified, data from all nodes in the cluster is displayed.

## Examples

This example shows sample output from the show ip ospf command.

```
device# show ip ospf
  vrf testname
OSPF Version                Version 2
Router Id                   0.0.0.0
ASBR Status                 No
ABR Status                  No           (0)
Redistribute Ext Routes from
Initial SPF schedule delay  0           (msecs)
Minimum hold time for SPF  5000        (msecs)
Maximum hold time for SPF  10000       (msecs)
External LSA Counter       0
External LSA Checksum Sum  0
Originate New LSA Counter  0
Rx New LSA Counter        0
External LSA Limit        14913080
Database Overflow Interval  0
Database Overflow State :  NOT OVERFLOWED
RFC 1583 Compatibility :   Enabled
NSSA Translator:          Enabled
Nonstop Routing:          Disabled
Originating router-LSAs with maximum metric
Condition: Always Current State: Active
Link Type: TRANSIT
Additional LSAs originated with maximum metric:
LSA Type      Metric Value
AS-External   16711680
Type 3 Summary 16711680
Type 4 Summary 16711680
```

This example shows sample output from the show ip ospf command, including BFD configuration information.

```
device# show ip ospf

OSPF Version                Version 2
Router Id                   19.19.19.19
ASBR Status                 No
ABR Status                  No           (0)
Redistribute Ext Routes from
Initial SPF schedule delay  0           (msecs)
Minimum hold time for SPF  0           (msecs)
Maximum hold time for SPF  0           (msecs)
External LSA Counter       0
External LSA Checksum Sum  00000000
Originate New LSA Counter  4
Rx New LSA Counter        5
External LSA Limit        14447047
Database Overflow Interval  0
Database Overflow State :  NOT OVERFLOWED
RFC 1583 Compatibility :   Enabled
Slow neighbor Flap-Action : Disabled,   timer 300
Nonstop Routing:          Disabled
Graceful Restart:         Disabled,   timer 120
Graceful Restart Helper:  Enabled
BFD:                      Enabled
LDP-SYNC: Not globally enabled
Interfaces with LDP-SYNC enabled: None
```

## History

Release version	Command history
6.0.1	This command was modified to include BFD configuration status.

## Related Commands

bfd, bfd holdover-interval, ip ospf bfd, show ip ospf neighbor

# show ip ospf area

Displays the OSPF area table in a specified format.

## Syntax

```
show ip ospf area { A.B.C.D | decimal } database link-state [ advertise index | asbr { asbrid | adv-router rid } | extensive | link-
state-id lid | network { netid | adv-router rid } | nssa { nsaaid | adv-router rid } | router { routerid | adv-router rid } | router-id
rid | self-originate | sequence-number num | summary { lid | adv-router rid } ] [ [ vrf vrfname [ rbridge-id { rbridge-id |
all } ] ] | rbridge-id { rbridge-id | all } ]
```

## Parameters

*A.B.C.D*

Area address in dotted decimal format.

*decimal*

Area address in decimal format. Valid values range from 0 to 2147483647.

**database link-state**

Displays database link-state information.

**advertise** *index*

Displays the link state by Link State Advertisement (LSA) index.

**asbr**

Displays the link state for all autonomous system boundary router (ASBR) links.

*asbrid*

Displays the state of a single ASBR link that you specify.

**adv-router** *rid*

Displays the link state for the advertising router that you specify.

**extensive**

Displays detailed information for all entries in the OSPF database.

**link-state-id** *lid*

Displays the link state by link-state ID.

**network**

Displays the link state by network link.

*netid*

Displays the link state of a particular network link that you specify.

**adv-router** *rid*

Displays the link state by the advertising router that you specify.

**nssa**

Displays the link state by not-so-stubby area (NSSA).

*nsaaid*

Displays the link state of a particular NSAA area that you specify.

**adv-router *rid***

Displays the link state for the advertising router that you specify.

**router**

Displays the link state by router link.

*routerid*

Displays the link state of a particular router link that you specify.

**adv-router *rid***

Displays the link state by the advertising router that you specify.

**router-id *rid***

Displays the link state by advertising router that you specify.

**self-originate**

Displays self-originated link states.

**sequence-number *num***

Displays the link-state by sequence number that you specify.

**summary**

Displays the link state summary. Can specify link-state ID or advertising router ID.

**adv-router *rid***

Displays the link state for the advertising router that you specify.

**vrf vrf *name***

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

```
device# show ip ospf area
Number of Areas is 3, NSSA area 2
```

Index	Area	Type	Cost	SPFR	ABR	ASBR	LSA	Chksum(Hex)	Translator
1	0	normal	0	5	0	1	7	000277c7	--
2	5	nssa	15	3	0	0	5	0004237d	Enabled
3	11	nssa	22	0	0	0	0	00000000	Candidate

show ip ospf area

## History

Release version	Command history
7.0.1	Command output was modified to include NSSA translator configuration information.



# show ip ospf border-routers

Displays information about border routers and boundary routers.

## Syntax

```
show ip ospf border-routers [ A.B.C.D ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ] | rbridge-id { rbridge-id | all }
```

## Parameters

*A.B.C.D*

Specifies the router ID in dotted decimal format.

**vrf** *vrf name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display information about area border routers (ABRs) and autonomous system boundary routers (ASBRs). You can display information for all ABRs and ASBRs or for a specific router.

## Examples

The following example displays information for all ABRs and ASBRs:

```
device# show ip ospf border-routers
```

Index	Router-ID	Router-type	Next-hop-router	Outgoing-interface	Area
1	1.0.0.1	ABR	22.22.22.2	2/2	7
1	1.0.0.2	ABR	22.22.22.2	2/2	7
1	1.0.0.1	ASBR	22.22.22.2	2/2	7
1	1.0.0.2	ASBR	22.22.22.2	2/2	7

# show ip ospf config

Displays OSPF configuration.

## Syntax

```
show ip ospf config
```

```
show ip ospf config ip-address { in | out } [ rbridge-id { rbridge-id | all } ] [ vrf vrf-name ]
```

```
show ip ospf config decimal { in | out } [ rbridge-id { rbridge-id | all } ] [ vrf vrf-name ]
```

```
show ip ospf config rbridge-id { rbridge-id | all }
```

```
show ip ospf config vrf vrf-name [ rbridge-id { rbridge-id | all }
```

## Parameters

*ip-address*

Specifies the IP address of an area.

**in**

Specifies the incoming direction.

**out**

Specifies the outgoing direction.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF instance are shown in the output.

*decimal*

Specifies an area address in decimal format. Valid values range from 0 through 2147483647.

## Modes

Privileged EXEC mode

## Examples

The following example displays information about OSPFv2 configurations.

```
device# show ip ospf config

Router OSPF: Enabled
Nonstop Routing: Disabled
Graceful Restart: Enabled
Graceful Restart Helper: Enabled
Graceful Restart Time: 120

Redistribution: Disabled
Default OSPF Metric: 10
Maximum Paths: 8
OSPF Auto-cost Reference Bandwidth: Disabled
Default Passive Interface: Disabled
OSPF Redistribution Metric: Type2
OSPF External LSA Limit: 14913080
OSPF Database Overflow Interval: 0
RFC 1583 Compatibility: Enabled
VRF Lite capability: Disabled
Router id: 10.10.10.2
OSPF Area currently defined:
Area-ID          Area-Type Cost
0                 normal  0
1                 normal  0
2.2.2.2          normal  0

OSPF Area Range currently defined:
Area-ID          Range-Address Subnetmask Status Config-Cost
1                15.15.15.0   255.255.255.0 advertise 45

OSPF type3 filters currently defined:
Area-ID          IN prefix OUT prefix
0                prefix1
1                prefix3
2.2.2.2          prefix4

OSPF Virtual Link Interfaces currently defined:
area 1           virtual-link 56.56.56.56 dead-int 40
area 1           virtual-link 56.56.56.56 hello-int 10
area 1           virtual-link 56.56.56.56 retransmit-int 5
area 1           virtual-link 56.56.56.56 transmit-delay 1
```

## History

Release version	Command history
7.0.0	This command was modified to display information about type 3 LSA filters configured for OSPFv2 areas.

# show ip ospf database

Shows database information.

## Syntax

```
show ip ospf database [ [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ] | rbridge-id { rbridge-id | all } ]
```

```
show ip ospf database database-summary [ [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ] | rbridge-id { rbridge-id | all } ]
```

```
show ip ospf database external-link-state [ advertise index | extensive | link-state-id lid | router-id routerid | sequence-number num ] [ [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ] | rbridge-id { rbridge-id | all } ]
```

```
show ip ospf database link-state [ advertise index | asbr { asbrid | adv-router rid } | extensive | link-state-id lid | network { netid | adv-router rid } | nssa { nsaaid | adv-router rid } | router [ { routerid | adv-router rid } | router-id routerid | self-originate | sequence-number num | summary { lid | adv-router rid } ]
```

## Parameters

### database-summary

Displays how many link state advertisements (LSAs) of each type exist for each area, as well as total number of LSAs.

### external-link-state

Displays information by external link state, based on the following parameters:

#### advertise *index*

Displays the hexadecimal data in the specified LSA packet. The *index* parameter identifies the LSA packet by its position in the router's External LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf external-link-state** command.

#### extensive

Displays LSAs in decrypt format. Do not use this parameter in combination with other display parameters because the entire database is displayed.

#### link-state-id *lid*

Displays external LSAs for the LSA source that you specify.

#### router-id *routerid*

Displays external LSAs for the advertising router that you specify.

#### sequence-number *num*

Displays the External LSA entries for the hexadecimal LSA sequence number that you specify.

### link-state

Displays the link state, based on the following parameters:

#### advertise *index*

Displays the hexadecimal data in the specified LSA packet. The *index* parameter identifies the LSA packet by its position in the router's external-LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf external-link-state** command.

#### asbr

Displays autonomous system boundary router (ASBR) LSAs.

**extensive**

Displays LSAs in decrypt format. Do not use this parameter in combination with other display parameters because the entire database is displayed.

**link-state-id** *lid*

Displays LSAs for the LSA source that you specify.

**network**

Displays either all network LSAs or the LSAs for a network that you specify.

**nssa**

Displays either all NSSA LSAs or the LSAs for a not-so-stubby area (NSSA) that you specify.

**router**

Displays LSAs by router link.

**router-id** *routerid*

Displays LSAs for the advertising router that you specify.

**self-originate**

Displays self-originated LSAs.

**sequence-number**

Displays the LSA entries for the hexadecimal LSA sequence number that you specify.

**summary**

Displays summary information. You can specify link-state ID or advertising router ID.

**adv-router** *rid*

Displays the link state for the advertising router that you specify.

**vrf** *name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

show ip ospf database

## Examples

```
switch# show ip ospf database
```

```
Link States
Index Area ID      Type LS ID          Adv Rtr             Seq(Hex) Age  Cksum
1      6                Summ 0.0.0.0         22.22.22.1        80000002 1   0xbfec
2      7                Rtr  22.22.22.1      22.22.22.1        80000002 6   0xb8cc
3      0                Summ 22.22.22.0       22.22.22.1        80000001 6   0x4294
```

```
switch# show ip ospf database
```

```
Link States
Index Area ID      Type LS ID          Adv Rtr             Seq(Hex) Age  Cksum
1      6                Summ 0.0.0.0         22.22.22.1        80000002 52  0xbfec
2      7                Rtr  22.22.22.1      22.22.22.1        80000003 7   0xda66
3      7                Rtr  1.0.0.2         1.0.0.2           80000001 1248 0xee99
4      7                Rtr  192.0.0.1       192.0.0.1         80000006 8   0x9c80
5      7                Rtr  1.0.0.1         1.0.0.1           80000001 1248 0xfe8b
6      7                Net  22.22.22.1     22.22.22.1        80000002 7   0xb419
7      7                Summ 1.0.2.0         1.0.0.1           80000001 1248 0x4314
8      7                Summ 1.0.0.0         1.0.0.1           80000001 1248 0x59ff
9      7                Summ 1.0.3.0         1.0.0.2           80000001 1248 0x3223
10     7                Summ 1.0.1.0         1.0.0.1           80000001 1248 0x4e0a
11     7                Summ 1.0.4.0         1.0.0.2           80000001 1248 0x272d
12     0                Summ 22.22.22.0     22.22.22.1        80000001 57  0x4294
13     0                ASBR 1.0.0.2         22.22.22.1        80000001 7   0x38db
14     0                ASBR 1.0.0.1         22.22.22.1        80000001 7   0x42d2

Type-5 AS External Link States
Index Age  LS ID      Router             Netmask Metric  Flag Fwd Address
1      1248 1.0.5.0    1.0.0.1           ffffffff00 00000001 0000 0.0.0.0
2      1248 1.0.8.0    1.0.0.2           ffffffff00 00000001 0000 0.0.0.0
3      1248 1.0.6.0    1.0.0.1           ffffffff00 00000001 0000 0.0.0.0
4      1248 1.0.7.0    1.0.0.2           ffffffff00 00000001 0000 0.0.0.0
```

# show ip ospf filtered-lsa area

Displays information about type3 LSA filters attached to specified OSPFv2 areas and lists LSAs filtered in or out.

## Syntax

```
show ip ospf filtered-lsa area { ip-address | decimal } { in | out } [ rbridge-id { rbridge-id | all } ] [ vrf vrf-name ]
```

## Parameters

*ip-address*

Specifies the IP address of an area.

*decimal*

Specifies an area address in decimal format. Valid values range from 0 through 2147483647.

**in**

Specifies the incoming direction.

**out**

Specifies the outgoing direction.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF instance are shown in the output.

## Modes

Privileged EXEC mode

## Examples

The following example displays information about type 3 LSA filtering in the out direction for OSPFv2 area 0.

```
device# show ip ospf filtered-lsa area 0 out
```

```
Prefix List Name: p3
Direction: OUT
Area: 0
Filtered LSA list:
Prefix      Mask
10.12.12.0  255.255.255.0
10.32.32.0   255.255.255.0
10.35.35.0   255.255.255.0
Total number of Filtered LSA :3
```

show ip ospf filtered-lsa area

The following example displays sample output for the **show ip ospf filtered-lsa area** command when the device is not an ABR.

```
device# do show ip ospf filtered-lsa area 0 out
Not an Area Border Router
device#
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show ip ospf interface

Displays information about all or specific OSPF-enabled interfaces.

## Syntax

```
show ip ospf interface [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
show ip ospf interface { A.B.C.D | brief } [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
show ip ospf interface { loopback number | port-channel number | ve vlan_id } [ brief ] [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
show ip ospf interface <N>gigabitethernet [ rbridge-id / ] slot / port [ brief ] [ vrf vrf-name ]
```

## Parameters

**vrf** *vrf-name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

*A.B.C.D*

Specifies interface IP address in dotted decimal format.

**brief**

Displays brief summary information about the specified port.

**loopback** *number*

Specifies a loopback port number. The range is from 1 through 255.

**port-channel** *number*

Specifies a port-channel interface. The range is from 1 through 6144.

**ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

(Optional) Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the physical interface type and name are specified, the **rbridge-id** option is not available.

## Examples

The following example displays OSPF information about all enabled interfaces.

```
device# show ip ospf interface
TenGigabitEthernet 3/0/1 admin up, oper up
IP Address 100.1.1.1, Area 0
Database Filter: Not Configured
State passive(default none), Pri 1, Cost 1, Options 2, Type broadcast Events 0
Timers(sec): Transmit 1, Retrans 5, Hello 10, Dead 40
DR: Router ID 0.0.0.0 Interface Address 0.0.0.0
BDR: Router ID 0.0.0.0 Interface Address 0.0.0.0
Neighbor Count = 0, Adjacent Neighbor Count= 0
Authentication-Key: None
MD5 Authentication: Key None, Key-Id None , Auth-change-wait-time 300
```

The following example displays OSPF information about a specified port-channel.

```
device# show ip ospf interface port-channel 10
No ospf interface entries available for specified command
```

# show ip ospf neighbor

Displays OSPF neighbor information.

## Syntax

```
show ip ospf neighbor [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
show ip ospf neighbor [ extensive ] [ port-channel number | router-id A.B.C.D | ve vlan_id ] [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

```
show ip ospf neighbor [ extensive ] [ <N>gigabitethernet [ rbridge-id / / slot / port ] [ vrf vrf-name ]
```

## Parameters

### vrf *vrf-name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF instance are shown in the output.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge.

#### all

Specifies all RBridges.

### extensive

Displays detailed information about all neighbors.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

### router-id *A.B.C.D*

Displays neighbor information for the specified router ID (in dotted decimal format).

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

### <*N*>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <*N*>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

(Optional) Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

If the physical interface type and name are specified, the **rbridge-id** option is not available.

## Examples

The following example displays information about all OSPF neighbors.

```
device# show ip ospf neighbor

Number of Neighbors is 2, in FULL state 1
Port  Address      Pri State      Neigh Address  Neigh ID      Ev Opt Cnt
2/2   22.22.22.1      0  FULL/OTHER  22.22.22.2    192.0.0.1     5  2  0
```

The following example shows detailed information about OSPF neighbors, including BFD configuration status.

```
device# show ip ospf neighbor extensive
Number of Neighbors is 1, in FULL state 1
Port  Address      Pri State      Neigh Address  Neigh ID      Ev      Opt Cnt
Ve 10   10.10.10.1    1  FULL/DR      10.10.10.2    2.2.2.2       6       2  0
Neighbor is known for 0d:00h:01m:34s and up for 0d:00h:00m:55s
Neighbor BFD State:UP, BFD HoldoverInterval(sec):Configured:2 Current:0
```

The following example shows detailed information about OSPF neighbors on a specified port-channel.

```
device# show ip ospf neighbor port-channel 10
No ospf neighbor entries available
```

## History

Release version	Command history
6.0.1	This command was modified to include BFD configuration status.
7.0.0	This command was modified to support port-channels.

## Related Commands

[bfd](#), [bfd holdover-interval](#), [ip ospf bfd](#), [show ip ospf](#)

# show ip ospf redistribute route

Displays routes that have been redistributed into OSPF.

## Syntax

```
show ip ospf redistribute route [A.B.C.D:M][[vrf vrfname [rbridge-id {rbridge-id | all}]]|rbridge-id {rbridge-id | all}]
```

## Parameters

*A.B.C.D:M*

Specifies an IP address and mask for the output.

**vrf** *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

```
switch# show ip ospf redistribute route
30.30.30.0 255.255.255.0 fwd 0.0.0.0 (0) metric 10 connected
50.1.0.0 255.255.0.0 fwd 100.1.1.100 (1) metric 10 static
```

# show ip ospf routes

Displays OSPF calculated routes.

## Syntax

```
show ip ospf routes [A.B.C.D][[vrf vrfname [rbridge-id {rbridge-id | all}]]|rbridge-id {rbridge-id | all}]
```

## Parameters

*A.B.C.D*

Specifies a destination IP address in dotted decimal format.

**vrf** *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display routes that OSPF calculated. You can display all routes or you can display information about a specific route.

## Examples

To display all OSPF-calculated routes:

```
switch# show ip ospf routes
```

```
OSPF Regular Routes 2:
  Destination      Mask          Path_Cost  Type2_Cost  Path_Type
  2.2.2.0          255.255.255.0  1          0           Intra
  Adv_Router      Link_State    Dest_Type  State       Tag        Flags
  2.2.2.1          2.2.2.1      Network   Valid       0          4000*
  Paths Out_Port  Next_Hop     Type       State
  1 eth 1/2        0.0.0.0     OSPF      00 00
  Destination      Mask          Path_Cost  Type2_Cost  Path_Type
  22.22.22.0       255.255.255.0  1          0           Intra
  Adv_Router      Link_State    Dest_Type  State       Tag        Flags
  2.2.2.1          22.22.22.1   Network   Valid       0          4000*
  Paths Out_Port  Next_Hop     Type       State
  1 eth 2/2        0.0.0.0     OSPF      00 00
```

# show ip ospf summary

Displays summary information for all OSPF instances.

## Syntax

```
show ip ospf summary [[ vrf vrfname [ rbridge-id { rbridge-id | all } ] ] | rbridge-id { rbridge-id | all } ]
```

## Parameters

**vrf** *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

```
device# show ip ospf summary
```

```
Total number of OSPF instances: 1
```

Seq	Instance	Intfs	Nbrs	Nbrs-Full	LSAs	Routes
1	default-vrf	5	2	1	12	2

# show ip ospf traffic

Displays OSPF traffic details.

## Syntax

**show ip ospf traffic**

**show ip ospf traffic** [ **loopback** *number* | **port-channel** *number* | **ve** *vlan\_id* ] [ **vrf** *vrf-name* ] [ **rbridge-id** { *rbridge-id* | **all** } ]

**show ip ospf traffic** <*N*>**gigabitethernet** [ *rbridge-id* / ] *slot* / *port* [ **vrf** *vrf-name* ]

## Parameters

**loopback** *number*

Specifies a loopback port number. The range is from 1 through 255.

**port-channel** *number*

Specifies a port-channel interface. The range is from 1 through 6144.

**ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

**vrf** *vrf-name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

<*N*>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <*N*>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

(Optional) Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode



## Usage Guidelines

Use this command to display details of OSPF traffic sent and received. You can display all traffic or specify a particular interface.

If the physical interface type and name are specified, the rbridge-id option is not available.

## Examples

The following example shows all OSPF traffic.

```
device# show ip ospf traffic
          Packets Received      Packets Sent
Hello                10                10
Database             90                89
LSA Req              12                11
LSA Upd              12                12
LSA Ack              12                12
No Packet Errors!
```

The following example shows OSPF traffic on a specified port-channel.

```
device# show ip ospf traffic port-channel 10
          Packets Received      Packets Sent
Hello                0                0
Database             0                0
LSA Req              0                0
LSA Upd              0                0
LSA Ack              0                0
No Packet Errors!
```

# show ip ospf virtual

Displays information about virtual links.

## Syntax

```
show ip ospf virtual { link | neighbor } [ index ] [ [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ] | rbridge-id { rbridge-id | all } ]
```

## Parameters

**link** *index*

Shows information about all virtual links or one virtual link that you specify.

**neighbor** *index*

Shows information about all virtual neighbors or one virtual neighbor that you specify.

**vrf** *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display information about virtual links and virtual neighbors over virtual links. You can show information about all virtual links or virtual neighbors, or you can specify a specific virtual link or virtual neighbor.

## Examples

To show information about all virtual links:

```
switch# show ip ospf virtual link
Indx      Transit Area      Router ID      Transit(sec) Retrans(sec) Hello(sec)
1
192.0.0.1      1
5            10
Dead(sec) events state Authentication-Key
40 0 down None
MD5 Authentication-Key: None
MD5 Authentication-Key-Id: None
MD5 Authentication-Key-Activation-Wait-Time: 300
```

To show information about all virtual neighbors:

```
switch# show ip ospf virtual neighbor
Indx      Transit Area      Router ID      Transit(sec)      Retrans(sec)      Hello(sec)
1
192.0.0.1      1      7
5
Dead(sec) events state Authentication-Key      10
40 0 down None
MD5 Authentication-Key: None
MD5 Authentication-Key-Id: None
MD5 Authentication-Key-Activation-Wait-Time: 300
```

# show ip pim bsr

Displays the bootstrap router (BSR) information.

## Syntax

```
show ip pim bsr [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Filter by RBridge ID.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

The information displayed ignores whether the Protocol Independent Multicast router is the elected BSR or not.

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays typical output for the **show ip pim bsr** form of this command.

```
device# show ip pim bsr

PIMv2 Bootstrap information :
-----
BSR address: 10.10.10.1. Hash Mask Length 32. Priority 0.
```

The following example displays BSR information for all RBridges in the cluster.

```
device# show ip pim bsr rbridge-id all

=====
Rbridge-id :1
=====
PIMv2 Bootstrap information :
-----
BSR address: 1.1.1.50. Hash Mask Length 30. Priority 1.

=====
Rbridge-id :2
=====
PIMv2 Bootstrap information :
-----
BSR address: 1.1.1.50. Hash Mask Length 30. Priority 1.
```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.

# show ip pim group

Displays the list of multicast groups.

## Syntax

```
show ip pim group [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Filter by RBridge ID.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display a list of the multicast groups that Protocol Independent Multicast (PIM) has learned. All groups, irrespective of how PIM learned them, are displayed.

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays the output for **show ip pim group**.

```
device# show ip pim group

Total number of groups: 2
1  Group 225.0.0.1
   Group member at   Te 19/2/1: Te 19/2/1
   Group member at   Ve 100: Ve 100
2  Group 225.0.0.2
   Group member at   Te 19/2/1: Te 19/2/1
   Group member at   Ve 100: Ve 100
```

The following example displays a list of multicast groups for all R Bridges in the cluster.

```
device# show ip pim group rbridge-id all
```

```
=====
Rbridge-id :1
=====
```

```
Total number of groups: 1
```

```
1   Group 230.2.2.2
    Group member at   Te 1/2/2: Te 1/2/2
```

```
=====
Rbridge-id :2
=====
```

```
Total number of groups: 1
```

```
1   Group 230.1.1.1
    Group member at   Te 2/2/1: Te 2/2/1
```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.

# show ip pim mcache

Displays the multicast cache.

## Syntax

```
show ip pim mcache [ ip-address-1 [ ip-address-2 ] ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*ip-address-1*

Group/Source IP address

*ip-address-2*

Group/Source IP address

**rbridge-id**

Filter by RBridge ID.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays the output for **show ip pim mcache ip-address-1 ip-address-2**.

```
device# show ip pim mcache 50.1.1.101 230.1.1.1
IP Multicast Mcache Table
Entry Flags      : sm - Sparse Mode, ssm - Source Specific Multicast
                  RPT - RPT Bit, SPT - SPT Bit, LSrc - Local Source
                  LRcv - Local Receiver, RegProbe - Register In Progress
                  RegSupp - Register Suppression Timer, Reg - Register Complete
                  needRte - Route Required for Src/RP
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert
                  MJ - Membership Join, BR - Blocked RPT, BA - Blocked Assert
                  BF - Blocked Filter
Total entries in mcache: 8
1 (50.1.1.101, 230.1.1.1) in Ve 40, Uptime 00:03:29
  Sparse Mode, RPT=0 SPT=1 Reg=0 RegSupp=0 RegProbe=0 LSrc=0 LRcv=1
  upstream neighbor=40.1.1.3
  num_oifs = 2
    Ve 2(00:03:29/181) Flags: IM
    Ve 10(00:03:29/0) Flags: MJ
  Flags (0x400784d1)
    sm=1 ssm=0 needRte=0
```



The following example displays the multicast cache summary for all R Bridges in the cluster.

```
device# show ip pim mcache rbridge-id all

=====
Rbridge-id :1
=====
IP Multicast Mcache Table
Entry Flags      : sm - Sparse Mode, ssm - Source Specific Multicast
                  RPT - RPT Bit, SPT - SPT Bit, LSrc - Local Source
                  LRcv - Local Receiver, RegProbe - Register In Progress
                  RegSupp - Register Suppression Timer, Reg - Register Complete
                  needRte - Route Required for Src/RP
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert
                  MJ - Membership Join, BR - Blocked RPT, BA - Blocked Assert
                  BF - Blocked Filter
Total entries in mcache: 2
1  (*, 230.1.1.1) RP 1.1.1.50 in Te 1/2/1, Uptime 00:47:32
   Sparse Mode, RPT=1 SPT=0 Reg=0 RegSupp=0 RegProbe=0 LSrc=0 LRcv=0
   Source is directly connected
   num_oifs = 1
   Ve 10(00:47:32/159) Flags: IM
   Flags (0x002204a0)
     sm=1 ssm=0 needRte=0
2  (*, 230.2.2.2) RP 1.1.1.50 in Te 1/2/1, Uptime 00:06:46
   Sparse Mode, RPT=1 SPT=0 Reg=0 RegSupp=0 RegProbe=0 LSrc=0 LRcv=1
   Source is directly connected
   num_oifs = 1
   Te 1/2/2(00:06:46/0) Flags: MJ
   Flags (0x012604a0)
     sm=1 ssm=0 needRte=0

=====
Rbridge-id :2
=====
IP Multicast Mcache Table
Entry Flags      : sm - Sparse Mode, ssm - Source Specific Multicast
                  RPT - RPT Bit, SPT - SPT Bit, LSrc - Local Source
                  LRcv - Local Receiver, RegProbe - Register In Progress
                  RegSupp - Register Suppression Timer, Reg - Register Complete
                  needRte - Route Required for Src/RP
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert
                  MJ - Membership Join, BR - Blocked RPT, BA - Blocked Assert
                  BF - Blocked Filter
Total entries in mcache: 1
1  (*, 230.1.1.1) RP 1.1.1.50 in Ve 10, Uptime 00:47:43
   Sparse Mode, RPT=1 SPT=0 Reg=0 RegSupp=0 RegProbe=0 LSrc=0 LRcv=1
   upstream neighbor=10.1.1.1
   num_oifs = 1
   Te 2/2/1(00:47:43/0) Flags: MJ
   Flags (0x012604a0)
     sm=1 ssm=0 needRte=0
```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.

# show ip pim neighbor

Displays the protocol independent multicast (PIM) neighbor information.

## Syntax

```
show ip pim neighbor [ interface <N>gigabitethernet rbridge-id / slot / port | port-channel number ]
```

```
show ip pim neighbor interface ve vlan-id [ rbridge-id { rbridge-id | all } ]
```

```
show ip pim neighbor rbridge-id { rbridge-id | all }
```

## Parameters

### interface

Specifies an interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface (VLAN interface number). The range is from 1 through 8191.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge.

#### all

Specifies all RBridges.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display information about all the neighbors that the PIM router perceives as active.

The **show ip pim neighbor** version of this command (without specifying **rbridge-id**) displays PIM neighbor information for the node on which the command is executed.

The **show ip pim neighbor interface ve** version of this command (without specifying **rbridge-id**) displays PIM neighbor information for the VE interface on the node on which the command is executed.

A neighbor for which join-suppression is enabled—by default or by running the **reset-tracking-bit** command—enables join-suppression for all of its neighbors.

## Examples

The following example displays the output for **show ip pim neighbor**.

```
device# show ip pim neighbor
Total Number of Neighbors : 43
Port      Phy_Port  Neighbor  Holdtime  Age  UpTime      Priority  Tracking-Bit
          sec    sec      Dd HH:MM:SS
Te 125/1/31 Te 125/1/31 125.31.1.1 105    0    00:29:30   1    Enabled
Te 125/1/43 Te 125/1/43 125.49.43.2 105    20   00:29:30   1    Enabled
Te 125/2/1  Te 125/2/1  125.2.1.2  105    10   00:29:40   1    Enabled
Ve 2000     Ve 2000     10.1.1.5   105    0    00:27:00   1    Enabled
Ve 2001     Ve 2001     21.1.1.3   105    0    00:27:00   1    Enabled
Ve 2002     Ve 2002     22.1.1.131 105    0    00:27:00   1    Enabled
```

The following example displays PIM neighbor information for all RBridges.

```
=====
Rbridge-id :1
=====
Total Number of Neighbors : 2
Port      Phy_Port  Neighbor  Holdtime  Age  UpTime      Priority  Tracking-Bit
          sec    sec      Dd HH:MM:SS
Te 1/0/4   Te 1/0/4   23.23.26.9 105    9    00:15:13   1    Enabled
Ve 10      Ve 10      23.23.27.6 105    3    00:24:07   1    Enabled

=====
Rbridge-id :2
=====
Total Number of Neighbors : 2
Port      Phy_Port  Neighbor  Holdtime  Age  UpTime      Priority  Tracking-Bit
          sec    sec      Dd HH:MM:SS
Te 2/0/4   Te 2/0/4   23.23.26.11 105    23   00:15:26   1    Enabled
Ve 10      Ve 10      23.23.27.5  105    3    00:23:38   1    Enabled
```

The following example displays PIM neighbor information for all RBridges on VE 10.

```
device# show ip pim neighbor int ve 10 rbridge-id all

=====
Rbridge-id :1
=====
Total Number of Neighbors : 1
Port      Phy_Port  Neighbor  Holdtime  Age  UpTime      Priority  Tracking-Bit
          sec    sec      Dd HH:MM:SS
Ve 10      Ve 10      23.23.27.6 105    1    00:24:35   1    Enabled

=====
Rbridge-id :2
=====
Total Number of Neighbors : 1
Port      Phy_Port  Neighbor  Holdtime  Age  UpTime      Priority  Tracking-Bit
          sec    sec      Dd HH:MM:SS
Ve 10      Ve 10      23.23.27.5  105    0    00:24:05   1    Enabled
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channel.
7.1.0	This command was modified to display secondary IP addresses when secondary IP addresses from PIM multinet enabled interfaces are received.

# show ip pim rpf

Displays the Reverse Path Forwarding (RPF) for a given unicast IP address.

## Syntax

```
show ip pim rpf A.B.C.D [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*A.B.C.D*

The unicast IP address.

**rbridge-id**

Filter by RBridge ID.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the port that PIM regards as the best reverse path for a given unicast IP address.

The unicast IP address may be an RP address or source address.

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays typical output for the **show ip pim rpf A.B.C.D** form of this command.

```
device# show ip pim rpf 123.32.120.10
Source 123.32.120.10 directly connected on Te 1/0/21
```

The following example displays the RPF for a specified unicast IP address, for all RBridges in the cluster.

```
device# show ip pim rpf 1.1.1.50 rbridge-id all
=====
Rbridge-id :1
=====
Source 1.1.1.50 directly connected on Te 1/2/1

=====
Rbridge-id :2
=====
upstream nbr 10.1.1.1 on Ve 10
```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.

# show ip pim rp-hash

Displays the Rendezvous Point (RP) information for a Protocol Independent Multicast (PIM) sparse group.

## Syntax

```
show ip pim rp-hash A.B.C.D [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*A.B.C.D*

Group address in dotted decimal format.

**rbridge-id**

Filter by RBridge ID.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays all RPs for the given group. The RP address could have been learned either from the Boot Strap Router (BSR) or configured statically.

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays typical output for the **show ip pim rp-hash A.B.C.D** form of this command.

```
device# show ip pim rp-hash 225.125.1.1
```

```
RP: 10.10.10.1, v2
```

The following example displays RP information for a PIM sparse group, for all RBridges in the cluster.

```
device# show ip pim rp-hash 230.1.1.1 rbridge-id all
```

```
=====
Rbridge-id :1
=====
```

```
RP: 1.1.1.50, v2
Info source: static RP configuration
```

```
=====
Rbridge-id :2
=====
```

```
RP: 1.1.1.50, v2
Info source: static RP configuration
```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.



# show ip pim rp-map

Displays the Rendezvous Point (RP) to group mappings.

## Syntax

```
show ip pim rp-map [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Filter by RBridge ID.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays typical output for the **show ip pim rp-map** form of this command.

```
switch# show ip pim rp-map

Number of group-to-RP mappings: 6
Group address RP address
-----
1 239.255.163.1 99.99.99.5
2 239.255.163.2 99.99.99.5
3 239.255.163.3 99.99.99.5
4 239.255.162.1 99.99.99.5
5 239.255.162.2 43.43.43.1
6 239.255.162.3 99.99.99.5
```

The following example displays RP-to-group mappings, for all RBridges in the cluster.

```

device# show ip pim rp-map rbridge-id all
=====
Rbridge-id :1
=====
Number of group-to-RP mappings: 2

      Group address      RP address
-----
1    230.1.1.1          1.1.1.50
2    230.2.2.2          1.1.1.50
=====
Rbridge-id :2
=====
Number of group-to-RP mappings: 1

      Group address      RP address
-----
1    230.1.1.1          1.1.1.50

```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.

# show ip pim rp-set

Displays the Rendezvous Point (RP) set list.

## Syntax

```
show ip pim rp-set [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Filter by RBridge ID.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays information regarding all RPs that Protocol Independent Multicast (PIM) perceives. The RPs may be either statically or dynamically learned.

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays typical output for the **show ip pim rp-set** form of this command.

```
device# show ip pim rp-set

Static RP
-----
Static RP count: 1
  RP: 22.22.22.22
Number of group prefixes Learnt from BSR: 1
Group prefix = 231.0.0.0/4      # RPs expected: 1
  # RPs received: 1
  RP 1: 33.33.33.33  priority=0   age=10   holdtime=150
```

The following example displays the RP set list, for all RBridges in the cluster.

```
device# show ip pim rp-set rbridge-id all
=====
Rbridge-id :1
=====
Static RP
-----
  Static RP count: 1

    RP: 1.1.1.50

=====
Rbridge-id :2
=====
Static RP
-----
  Static RP count: 1

    RP: 1.1.1.50
```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.

# show ip pim-sparse

Displays the internal parameters of the protocol independent multicast (PIM) router or the PIM enabled interface.

## Syntax

```
show ip pim-sparse [ interface <N>gigabitethernet rbridge-id / slot / port | port-channel number ]
```

```
show ip pim-sparse interface ve vlan-id [ rbridge-id { rbridge-id | all } ]
```

```
show ip pim-sparse rbridge-id { rbridge-id | all }
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel interface. Range is from 1 through 6144.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **show ip pim-sparse** version of this command (without specifying **rbridge-id**) displays information for the node on which the command is executed.

The **show ip pim-sparse interface ve** version of this command (without specifying **rbridge-id**) displays information for the VE interface on the node on which the command is executed.

## Examples

The following example displays the output for **show ip pim-sparse**.

```
device# show ip pim-sparse
Maximum mcache      : 0          Current count      : 0
Hello interval      : 30         Neighbor timeout   : 105
Join/Prune interval : 60         Inactivity interval : 180
Hardware drop enabled : Yes       Prune wait interval : 3
Register Suppress Time : 60       Register Probe Time : 10
Register Stop Delay  : 60         Register Suppress interval : 60
SSM Enabled         : No          SPT Threshold     : 1
Tracking Bit        : Enabled
```

The following example displays PIM information for all R Bridges in the cluster.

```
device# show ip pim-sparse rbridge-id all
```

```
=====
Rbridge-id :2
=====
Maximum mcache      : 2048       Current count      : 2
Hello interval      : 30         Neighbor timeout   : 105
Join/Prune interval : 60         Inactivity interval : 180
Hardware drop enabled : Yes       Prune wait interval : 3
Register Suppress Time : 60       Register Probe Time : 10
Register Stop Delay  : 60         Register Suppress interval : 10
SSM Enabled         : No          SPT Threshold     : 1
Tracking Bit        : Enabled

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Interface |Local |Ver |Mode |State| Designated Router |TTL|Multicast|Neighbor |
DR        |Address | | | | |Address |Port |Thr|Boundary |Filter |
Prio
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Te 2/0/30 23.1.1.2 v2 SM Up Itself 1 None None 1
Ve 35 35.1.1.1 v2 SM Up 35.1.1.2 Ve 35 1 None None 1
Ve 88 88.1.1.2 v2 SM Down Itself 1 None None 1
Lo 1 33.33.33.33 v2 SM Up Itself 1 None None 1

=====
Rbridge-id :3
=====
Maximum mcache      : 2048       Current count      : 0
Hello interval      : 30         Neighbor timeout   : 105
Join/Prune interval : 10         Inactivity interval : 180
Hardware drop enabled : Yes       Prune wait interval : 3
Register Suppress Time : 60       Register Probe Time : 10
Register Stop Delay  : 60         Register Suppress interval : 10
SSM Enabled         : No          SPT Threshold     : 1
Tracking Bit        : Enabled

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Interface |Local |Ver |Mode |State| Designated Router |TTL|Multicast|Neighbor |
DR        |Address | | | | |Address |Port |Thr|Boundary |Filter |
Prio
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Te 3/0/20 24.1.1.2 v2 SM Up Itself 1 None None 1
Ve 45 45.1.1.1 v2 SM Down Itself 1 None None 1
Ve 88 88.1.1.3 v2 SM Up Itself 1 None None 1
Lo 1 44.44.44.44 v2 SM Up Itself 1 None None 1
device#
```

The following example displays PIM information for all R Bridges on VE 10.

```
device# show ip pim-sparse interface ve 10 rbridge-id all
```

```
=====
Rbridge-id :1
=====
```

```
Maximum mcache      : 2048 Current count      : 0
Hello interval      : 30 Neighbor timeout     : 105
Join/Prune interval : 60 Inactivity interval  : 180
Hardware drop enabled : Yes Prune wait interval : 3
Register Suppress Time : 60 Register Probe Time : 10
Register Stop Delay  : 60 Register Suppress interval : 60
SSM Enabled         : No SPT Threshold       : 1
Tracking Bit        : Enabled
```

```
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Interface|Local      |Ver |Mode| Designated Router |TTL|Multicast|Neighbor | DR
         |Address   |   |   | Address          Port |Thr|Boundary |Filter  | Prio
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Ve 10    | 23.23.27.5 |v2SM SM | 23.23.27.6 | Ve 10 | 1 | None | None | 1
```

```
=====
Rbridge-id :2
=====
```

```
Maximum mcache      : 2048 Current count      : 0
Hello interval      : 30 Neighbor timeout     : 105
Join/Prune interval : 60 Inactivity interval  : 180
Hardware drop enabled : Yes Prune wait interval : 3
Register Suppress Time : 60 Register Probe Time : 10
Register Stop Delay  : 60 Register Suppress interval : 60
SSM Enabled         : No SPT Threshold       : 1
Tracking Bit        : Enabled
```

```
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Interface|Local      |Ver |Mode| Designated Router |TTL|Multicast|Neighbor | DR
         |Address   |   |   | Address          Port |Thr|Boundary |Filter  | Prio
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
Ve 10    | 23.23.27.6 |v2SM SM | Itself |      | 1 | None | None | 1
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channel.
7.1.0	A new column (State) introduced in the output.

# show ip pim traffic

Displays the Protocol Independent Multicast (PIM) traffic statistics categorized by each PIM enabled interface.

## Syntax

```
show ip pim traffic [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Filter by RBridge ID.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridge IDs.

## Modes

Privileged EXEC mode

## Usage Guidelines

If you do not include the **rbridge-id** keyword, this command displays output for the current node only.

## Examples

The following example displays typical output for the **show ip pim traffic** form of this command.

```
device# show ip pim traffic
Port      |HELLO      |JOIN        |PRUNE       |ASSERT      |GRAFT/REGISTER  |REGISTER-STOP
          |Rx         Tx          |Rx          Tx          |Rx           Tx          |Rx           Tx          |Rx           Tx          |Rx           Tx
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
Te 2/2/1  0         108         0           0           0           0           0           0           0           0           0           0
Ve 10     229       231         0           109         0           5           0           0           0           0           0           0
```



The following example displays PIM traffic statistics, for all RBridges in the cluster.

```
device# show ip pim traffic rbridge-id all
=====
Rbridge-id :1
=====
Port      |HELLO      |JOIN      |PRUNE      |ASSERT      |GRAFT/REGISTER|REGISTER-STOP
          |Rx   Tx   |Rx   Tx   |Rx   Tx   |Rx   Tx   |Rx   Tx   |Rx   Tx
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
Te 1/2/1  105  109  0    67   0    0  0    0    0  0    0    0    0
Te 1/2/2  0    27   0    0    0    0  0    0    0    0    0    0
Ve 10     230  233  109  0    5    0  0    0    0    0    0    0
=====
Rbridge-id :2
=====
Port      |HELLO      |JOIN      |PRUNE      |ASSERT      |GRAFT/REGISTER|REGISTER-STOP
          |Rx   Tx   |Rx   Tx   |Rx   Tx   |Rx   Tx   |Rx   Tx   |Rx   Tx
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
Te 2/2/1  0    109  0    0    0    0  0    0  0    0    0    0
Ve 10     230  231  0    109  0    5    0    0  0    0    0    0
=====
```

## History

Release version	Command history
6.0.1	The <b>rbridge-id all</b> option was added.

# show ip prefix-list

Displays the status of an IPv4 prefix list.

## Syntax

```
show ip prefix-list name [rbridge-id number] | rbridge-id list_name]
```

## Parameters

*name*

Name of an IPv4 prefix list.

**rbridge-id***number*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To display the status of the IPv4 prefix list "mylist":

```
device# show ip prefix-list mylist
```

# show ip route

Shows IP route information for numbered and unnumbered IP interfaces.

## Syntax

```
show ip route A.B.C.D [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route A.B.C.D/M [ longer ] [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route all [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route bgp [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route connected [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route detail [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route import [ src-vrf-name ] [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route nexthop [ nexthopID ] [ ref-routes ] [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route ospf [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route rbridge-id { rbridge-id | all } [ vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route slot line-card-number [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route static [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route summary [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ]
show ip route vrf vrf-name [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*A.B.C.D/M*

IPv4 address and optional mask.

**longer**

Specifies routes that match the specified prefix.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies routes for a selected VRF instance.

**all**

Specifies information for all configured IPv4 routes.

**bgp**

Specifies BGP route information.

**connected**

Specifies directly connected routes, such as local Layer 3 interfaces.

**detail**

Specifies detailed information about routes.

**import**

Specifies imported IPv4 routes.

*src-vrf-name*

Specifies a VRF instance from which routes are leaked.

**nexthop**

Specifies the configured next hop.

*nexthopID*

Valid values range from 0 through 4294967294.

**ref-routes**

Specifies all routes that point to the specified *nexthopID*.

**ospf**

Specifies routes learned from the Open Shortest Path First (OSPF) protocol.

**rbridge-id** *rbridge-id*

Specifies routes for a selected RBridge ID.

**slot** *line-card-number*

Specifies routes with the provided line card number.

**static**

Specifies configured static routes.

**summary**

Specifies summary information for all routes.

**system-summary**

Specifies summary information for IP routes.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

If leaked subnet routes are present, that information displays in the output.

To view the status of management routes, use the **show ip route vrf** command and enter **mgmt-vrf** as the VRF name. You must enter the name of the management VRF manually.

## Examples

The following example shows output for the standard output.

```
device# show ip route vrf mgmt-vrf
Total number of IP routes: 3
Type Codes - B:BGP D:Connected O:OSPF S:Static U:Unnumbered +:Leaked route; Cost - Dist/Metric
BGP Codes - i:iBGP e:eBGP
OSPF Codes - i:Inter Area 1:External Type 1 2:External Type 2 s:Sham Link
  Destination      Gateway          Port          Cost          Type Uptime
  0.0.0.0/0        10.25.224.1     mgmt 1        1/1           S   4d19h
  10.25.224.0/24   DIRECT          mgmt 1        0/0           D   4d19h
  10.25.224.18/32  DIRECT          mgmt 1        0/0           D   4d19h
```

The following example shows output for the **show ip route system-summary** command.

```
device# show ip route system-summary
System Route Count: 3 Max routes: 4096 (Route limit not exceeded)
System Nexthop Count: 2 Max nexthops: 1024 (Nexthop limit not exceeded)

VRF-Name: default-vrf
  Route count: 0 Max routes: Not Set (Route limit not exceeded)
  0 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered

VRF-Name: mgmt-vrf
  Route count: 3 Max routes: Not Set (Route limit not exceeded)
  1 connected, 1 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered
```

The following example shows output for the **connected** option.

```
device# show ip route connected
Type Codes - B:BGP D:Connected I:ISIS O:OSPF R:RIP S:Static; Cost - Dist/Metric
BGP Codes - i:iBGP e:eBGP
ISIS Codes - L1:Level-1 L2:Level-2
OSPF Codes - i:Inter Area 1:External Type 1 2:External Type 2 s:Sham Link
  Destination      Gateway          Port          Cost          Type Uptime
  1 1.1.1.0/24       DIRECT          Te 2/1        0/0           D   4m33s
  2 1.1.2.0/24       DIRECT          Te 2/2        0/0           D   2m42s
```

The following example shows output for the **summary** option.

```
device# show ip route summary
IP Routing Table - 5008 entries:
  4 connected, 0 static, 0 OSPF, 5000 BGP, 0 ISIS, 0 unnumbered, 0 EVPN Host
```

The following example shows output for the **slot <number> vrf <name>** option.

```
device# show ipv6 route slot 0 vrf red
IPv6 Routing Table for VRF "red"
'*' denotes best ucast next-hop
'[x/y]' denotes [preference/metric]
'Inactive' denotes not installed in hardware

101:1:1::/64, attached
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 7m11s, eBgp, (VNI 120, GW MAC 50EB:1AAC:2C41, Tu 61441)
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 7m11s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
101:1:1:1/128, attached, Inactive
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 7m11s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
101:1:1:2/128, attached, Inactive
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 7m11s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
101:1:1:3/128, attached, Inactive
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 7m11s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
102:1:1::/64, attached
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 8m8s, eBgp, (VNI 120, GW MAC 50EB:1AAC:2C41, Tu 61441)
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 8m8s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
102:1:1:1/128, attached, Inactive
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 8m8s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
102:1:1:2/128, attached, Inactive
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 8m8s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
102:1:1:3/128, attached, Inactive
  *via 32.0.0.32%default-vrf, Ve 120, [0/0], 8m8s, eBgp, (VNI 120, GW MAC 50EB:1AAA:30B1, Tu 61441)
```

The following example shows output for the **nexthop** option.

```
device# show ip route nexthop
Total number of IP nexthop entries: 4; Forwarding Use: 4
  NexthopIp      Port      RefCount  ID      Age
1      1.1.1.2      Te 2/1      3/3      2147549184 277
2      0.0.0.0      Te 2/2      1/1      2147484008 191
3      0.0.0.0      Te 2/1      2/2      2147484009 302
4      1.1.1.2      Te 2/1      1/1      2147549185 190
      1.1.2.2      Te 2/2
```

The following example shows output for a specific next hop.

```
device# show ip route nexthop 2147549184
  NexthopIp      Port      RefCount  ID      Age
1      1.1.1.2      Te 2/1      3/3      2147549184 288
```

The following example shows output for the **nexthop** option.

```
device# show ip route nexthop 2147549184 ref-routes
Type Codes - B:BGP D:Connected I:ISIS O:OSPF R:RIP S:Static; Cost - Dist/Metric
BGP Codes - i:iBGP e:eBGP
ISIS Codes - L1:Level-1 L2:Level-2
OSPF Codes - i:Inter Area 1:External Type 1 2:External Type 2 s:Sham Link
  Destination      Gateway      Port      Cost      Type Uptime
1      100.1.1.0/24      1.1.1.2      Te 2/1      1/1      S      5m10s
2      100.1.2.0/24      1.1.1.2      Te 2/1      1/1      S      4m54s
3      100.1.3.0/24      1.1.1.2      Te 2/1      1
```

The following example shows output for a specific IP address:

```
device# show ip route 100.1.1.1
Type Codes - B:BGP D:Connected I:ISIS O:OSPF R:RIP S:Static; Cost - Dist/Metric
BGP Codes - i:iBGP e:eBGP
ISIS Codes - L1:Level-1 L2:Level-2
OSPF Codes - i:Inter Area 1:External Type 1 2:External Type 2 s:Sham Link
  Destination      Gateway      Port      Cost      Type Uptime
4      100.1.1.0/24      1.1.1.2      Te 2/1      1/1      S      5m37s
```

The following example shows output for a specific IP address:

```
device# show ip route 4.5.1.0/24 detail
IP Routing Table for VRF "default-vrf"
Total number of IP routes: 18
'*' denotes best ucast next-hop
'[x/y]' denotes [preference/metric]

[*]4.5.1.0/24,
  via 3.4.0.3, Ve 3402, [20/0], 33m32s, eBgp, tag 0
  via 3.5.0.3, Ve 3502, [20/0], 33m32s, eBgp, tag 0
4.5.1.0/24,
  via DIRECT, Ve 3402, [250/1], 0m4s, static, tag 0
```

The following example shows output for the **longer** option.

```
device# show ip route 100.0.0.0/8 longer
1      100.1.1.0/24      1.1.1.2      Te 2/1      1/1      S      14m37s
2      100.1.2.0/24      1.1.1.2      Te 2/1      1/1      S      14m21s
3      100.1.3.0/24      1.1.1.2      Te 2/1      1/1      S      14m18s
4      100.2.1.0/24      DIRECT        Te 2/1      1/1      S      14m2s
5      100.3.1.0/24      1.1.1.2      Te 2/1      1/1      S      13m10s
      100.3.1.0/24      1.1.2.2      Te 2/2      1/1      S      13m10s
```

The following example shows output for the **detail** option.

```
device# show ip route detail
IP Routing Table for VRF "default-vrf"
Total number of IP routes: 18
'[x/y]' denotes [preference/metric]

1.0.0.0/31, attached
  via 5.0.0.0, Po 5, [20/0], 21h9m, eBgp, tag 0
1.0.1.0/31, attached
  via 5.0.0.0, Po 5, [20/0], 23h50m, eBgp, tag 0
1.1.1.1/32, attached
  via 5.0.0.0, Po 5, [20/0], 9m28s, eBgp, tag 0
  via 104.3.3.2, Po 104, [20/0], 9m28s, eBgp, tag 0
1.75.10.0/24, attached
  via 5.0.0.0, Po 5, [20/0], 23h50m, eBgp, tag 0
  via 104.3.3.2, Po 104, [20/0], 23h50m, eBgp, tag 0
2.2.2.2/32, attached
  via 5.0.0.0, Po 5, [20/0], 23h50m, eBgp, tag 0
```

The following example shows output for the **detail vrf** option.

```
device# show ip route detail vrf red
IP Routing Table for VRF "red"
'*' denotes best ucast next-hop
'[x/y]' denotes [preference/metric]

3.3.3.0/24, attached
  *via DIRECT, Te 89/0/8, [0/0], 0m11s, static+(default-vrf), tag 0
```

## History

Release version	Command history
6.0.0	This command was modified to support the <b>vrf</b> keyword.
6.0.1a	This command was modified to show examples for the <b>detail</b> keyword.
7.0.0	This command was modified to show examples for unnumbered IP interfaces.
7.0.1	This command was modified to display leaked subnet routes.
7.2.0	The command output display was enhanced.

# show ip route import

Displays the IPv4 routes imported to a specified VRF

## Syntax

```
show ip route import [vrf vrf_name] [rbridge-id {rbridge-id | all} ]
```

## Parameters

### **vrf\_name**

Specifies the VRF whose imported routes you want to display.

### **rbridge-id**

Specifies a RBridge or all RBridges for the VRF whose routes you wish to display.

#### *rbridge-id*

Specifies an RBridge ID.

#### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode



## Usage Guidelines

## Examples

To display IPv4 routes for a VRF that has been configured with the name VRF2:

```
switch# show ip route import vrf vrf2
Total number of IP routes: 106
Type Codes - B:BGP D:Connected O:OSPF S:Static; Cost - Dist/Metric
BGP Codes - i:iBGP e:eBGP
OSPF Codes - i:Inter Area 1:External Type 1 2:External Type 2 s:Sham Link
  Destination      Gateway           Port             Cost           Type Uptime
  12.0.0.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.1.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.2.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.3.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.4.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.5.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.6.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.7.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.8.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.9.0/24      10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.10.0/24     10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.11.0/24     10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.12.0/24     10.1.21.200     Ve 21            20/0           Be+  1m52s
  12.0.13.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
  12.0.14.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
  12.0.15.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
  12.0.16.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
  12.0.17.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
  12.0.18.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
  12.0.19.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
  12.0.20.0/24     10.1.22.200     Ve 22            20/0           Be+  1m52s
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ip route system-summary

Displays IPv4 route information with respect to route limits and next-hop limits, as well as additional information, for all VRFs and specific VRFs.

## Syntax

```
show ip route system-summary [ rbridge-id rbridge-id ] [ vrf name ]
```

## Parameters

**rbridge-id** *rbridge-id*

Displays routes for a selected RBridge ID.

**vrf** *name*

Displays routes for a selected VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Per-VRF data are as for the **show ip route summary vrf** command, except for all VRFs.

The **show ip route system-summary** command displays the following information:

Output field	Description
System Route Count	Displays current (all VRF aggregate) system-wide route count, the maximum value supported by the RBridge, and whether or not the system limit was exceeded in the past.
System Nexthop Count	Displays current (all VRF aggregate) system-wide next-hop count, the maximum value supported by the RBridge, and whether or not the system limit was exceeded in the past.
VRF-Name	Displays, on a per-VRF basis, the current route count on the VRF, the maximum number of routes (if configured) on the VRF, and a breakdown of routes as connected, static, per routing protocol, and so on. The display is for a named VRF, the default VRF, and the management VRF.

## Examples

Typical command output:

```
switch# show ip route system-summary

System Route Count: 72 Max routes: 6144 (Route limit not exceeded)
System Nexthop Count: 2 Max nexthops: 1024 (Nexthop limit not exceeded)
One Path Nexthop Count: 3 Max One Path Nexthops: 5120

VRF-Name: blue
  Route count: 16 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered, 0 EVPN Host

VRF-Name: default-vrf
  Route count: 5 Max routes: Not Set (Route limit not exceeded)
  3 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered, 0 EVPN Host

VRF-Name: green
  Route count: 16 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered, 0 EVPN Host

VRF-Name: mgmt-vrf
  Route count: 3 Max routes: Not Set (Route limit not exceeded)
  1 connected, 1 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered, 0 EVPN Host

VRF-Name: red
  Route count: 16 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered, 0 EVPN Host

VRF-Name: yellow
  Route count: 16 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 unnumbered, 0 EVPN Host
```

## History

Release version	Command history
5.0.1	This command was introduced.
7.0.1	One Path Nexthop Count option added.

# show ip route-map

Displays the status of an IPv4 route map.

## Syntax

```
show ip route-map name[rbridge-id number] | rbridge-id list_name ]
```

## Parameters

*name*

Name of an IPv4 route map.

**rbridge-id***number*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

To display the status of the IPv4 route map "myroutemap":

```
device# show ip prefix-list myroutemap
```

# show ip static route

Displays information related to IPv4 static routes.

## Syntax

```
show ip static [ A.B.C.D/M [ rbridge-id rbridge-id | vrf vrf-name ]
```

## Parameters

*A.B.C.D/M*

Specifies an IPv4 address and mask.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

To display the status of static routes for subnet 10.1.1.1/24 and VRF instance "red":

```
device# show ip static route 10.1.1.1/24 vrf red
```

# show ipv6 anycast-gateway

Displays IPv6 anycast-gateway details for all virtual Ethernets (VEs) or for a specified VE. You can also filter by RBridge and by VRF.

## Syntax

```
show ipv6 anycast-gateway [ interface ve vlan-id ] [ rbridge-id { rbridge-id | range | all } ]
```

```
show ipv6 anycast-gateway [ vrf { vrf-name | all } ] [ rbridge-id { rbridge-id | range | all } ]
```

## Parameters

**interface ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

**rbridge-id**

Specifies an RBridge, multiple RBridges, or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

*range*

Specifies multiple RBridge IDs. You can specify a range (for example, 3-5), a comma-separated list (for example, 1,3,5,6), or you can combine a range with a list (for example, 1-5,6,8). In a range string, no spaces are allowed.

**all**

Specifies all RBridges.

**vrf**

Specifies all VRF instances or one VRF instance. Without the **vrf** keyword, details for the default VRF instance are shown in the output.

*vrf-name*

Specifies a vrf name.

**all**

Specifies all vrfs.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 anycast-gateway** command displays the following information:

Output field	Description
Gateway mac	Displays the MAC address specified for the anycast gateway.
Interface	Displays the virtual Ethernet (VE) interface.

Output field	Description
ip address	Displays the IP anycast address and mask.
state	Displays "Active" or "Inactive". If the value is "Inactive", the reason is displayed in brackets.

## Examples

The following example displays IPv6 anycast-gateway information on all VEs.

```
device# show ipv6 anycast-gateway
Gateway mac: 000a.000b.000d
Interface    Ip address    state
ve10        1234:2::22/64  Active
ve10        1234:3::22/64  Active
ve20        1234:33::33/64  Active
```

The following example displays IPv6 anycast-gateway information on a specified VE.

```
device# show ipv6 anycast-gateway interface ve 10
Gateway mac: 000a.000b.000d
Interface    Ip address    state
ve10        1234:2::22/64  Active
ve10        1234:3::22/64  Active
```

The following example displays the default vrf anycast-gateway sessions.

```
device# show ipv6 anycast-gateway vrf default-vrf
Interface    Ip address    state
Gateway mac: 0000.aaaa.cccc

Ve10        1234:2::22/64  Active
ve10        1234:3::22/64  Active
ve20        1234:33::33/64  Active
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show ipv6 bgp attribute-entries

Displays BGP4+ route-attribute entries that are stored in device memory.

## Syntax

```
show ipv6 bgp attribute-entries [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf vrf-name**

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

The route-attribute entries table lists the sets of BGP4+ attributes that are stored in device memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the device typically has fewer attribute entries than routes. Use this command to view BGP4+ route-attribute entries that are stored in device memory.

## Examples

The following is sample output from the **show ipv6 bgp attribute-entries** command.

```
device# show ipv6 bgp attribute-entries

Total number of BGP Attribute Entries: 2
1      Next Hop   : 2001::1                               MED      :1          Origin:IGP
      Originator:0.0.0.0           Cluster List:None
      Aggregator:AS Number :0       Router-ID:0.0.0.0       Atomic:None
      Local Pref:1                 Communities:Internet
      AS Path   : (length 0)
      Address: 0x1205c75c Hash:268 (0x01000000)
      Links: 0x00000000, 0x00000000
      Reference Counts: 2:0:0, Magic: 1
2      Next Hop   : ::                                   MED      :1          Origin:IGP
      Originator:0.0.0.0           Cluster List:None
      Aggregator:AS Number :0       Router-ID:0.0.0.0       Atomic:None
      Local Pref:100                Communities:Internet
      AS Path   : (length 0)
      AsPathLen: 0 AsNum: 0, SegmentNum: 0, Neighboring As: 0, Source As 0
      Address: 0x1205c7cc Hash:365 (0x01000000)
      Links: 0x00000000, 0x00000000
      Reference Counts: 1:0:1, Magic: 2
```



## History

Release version	Command history
5.0.0	This command was introduced.
6..0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp dampened-paths

Displays all BGP4+ dampened routes.

## Syntax

```
show ipv6 bgp dampened-paths [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

The following is sample output from the **show ipv6 bgp dampened-paths** command.

```
device# show ipv6 bgp dampened-paths
```

	Network	From		Flaps	Since	Reuse	Path
*d	2001:db8:8::/45	2001:db8:1::1	1	0 :1 :14	0 :2 :20	100	1002 1000
*d	2001:db8:1::/48	2001:db8:1::1	1	0 :1 :14	0 :2 :20	100	1002 1000
*d	2001:db8:4::/46	2001:db8:1::1	1	0 :1 :14	0 :2 :20	100	1002 1000
*d	2001:db8:2::/47	2001:db8:1::1	1	0 :1 :14	0 :2 :20	100	1002 1000
*d	2001:db8:0:8000::/49	2001:db8:1::1	1	0 :1 :14	0 :2 :20	100	1002 1000
*d	2001:db8:17::/64	2001:db8:1::1	1	0 :1 :18	0 :2 :20	100	

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp filtered-routes

Displays BGP4+ filtered routes that are received from a neighbor or peer group.

## Syntax

```
show ipv6 bgp filtered-routes ipv6-addr mask [ longer-prefixes [ rbridge-id { rbridge-id | all } ] | rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp filtered-routes as-path-access-list name [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ]
```

```
show ipv6 bgp filtered-routes prefix-list name [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ]
```

```
show ipv6 bgp filtered-routes [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of the destination network in dotted-decimal notation.

*mask*

IPv6 mask of the destination network in CIDR notation.

**longer-prefixes**

Specifies all statistics for routes that match the specified route, or that have a longer prefix than the specified route.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

**as-path-access-list**

Specifies an AS-path ACL.

**name**

Name of ACL.

**prefix-list**

Specifies an IP prefix list.

**name**

Name of prefix-list

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command with the **longer-prefixes** keyword to display routes that match a specified or longer IPv6 prefix. For example, if you specify 2001:db8::/16 longer-prefixes , then all routes with the prefix 2001:db8::/16 or that have a longer prefix (such as 2001:db8::/32) are displayed.

## Command Output

The **show ipv6 bgp filtered-routes** command displays the following information.

Output field	Description
Number of BGP4+ Routes matching display condition	The number of routes that matched the display parameters you entered. This is the number of routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route's status. The status code appears in the left column of the display, to the left of each route. The status codes are described in the command's output. The status column displays an "IF" for each filtered route.
Prefix	The network address and prefix.
Next Hop	The next-hop router for reaching the network from the device.
MED	The value of the route's MED attribute. If the route does not have a metric, this field is blank.
LocPrf	The degree of preference for this route relative to other routes in the local AS. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295.
Weight	The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight.
Status	The route's status, which can be one or more of the following: <ul style="list-style-type: none"> <li>• A - AGGREGATE - The route is an aggregate route for multiple networks.</li> <li>• B - BEST - BGP4+ has determined that this is the optimal route to the destination.</li> <li>• b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3 or static IPv6 routes).</li> <li>• C - CONFED_EBGP - The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation.</li> <li>• D - DAMPED - This route has been dampened (by the route dampening feature), and is currently unusable.</li> <li>• E - EBGP - The route was learned through a in another AS.</li> <li>• H - HISTORY - Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.</li> <li>• I - IBGP - The route was learned through a in the same AS.</li> <li>• L - LOCAL - The route originated on this device.</li> <li>• M - MULTIPATH - BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".</li> </ul>

Output field	Description
	<p><b>NOTE</b> If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.</p> <ul style="list-style-type: none"> <li>• S - SUPPRESSED - This route was suppressed during aggregation and thus is not advertised to neighbors.</li> <li>• F - FILTERED - This route was filtered out by BGP4+ route policies on the device, but the device saved updates containing the filtered routes.</li> </ul>

## Examples

This example shows sample output from the **show ipv6 bgp filtered-routes** command when no keyword is used.

```
device# show ipv6 bgp filtered-routes
```

```
Searching for matching routes, use ^C to quit...
```

```
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
Prefix           Next Hop           MED LocPrf      Weight Status
1      2001:db8:3000::/48 2001:db8::110      100      0             EF
      AS_PATH: 65001 4355 701 80
2      2001:db8:4000::/48 2001:db8::110      100      0             EF
      AS_PATH: 65001 4355 1
3      2001:db8:5000::/48 2001:db8::110      100      0             EF
      AS_PATH: 65001 4355 701 1 189
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ipv6 bgp filtered-routes detail

Displays detailed information about routes that have been filtered out by BGP4+ route policies.

## Syntax

```
show ipv6 bgp filtered-routes detail ipv6-addr mask [ longer-prefixes [ rbridge-id { rbridge-id | all } ] | rbridge-id { rbridge-id | all } ]
show ipv6 bgp filtered-routes detail as-path-access-list name [ rbridge-id { rbridge-id | all } ]
show ipv6 bgp filtered-routes detail prefix-list name [ rbridge-id { rbridge-id | all } ]
show ipv6 bgp filtered-routes detail [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*ipv6-addr*

IPv6 address of the destination network in dotted-decimal notation.

*mask*

IPv6 mask of the destination network in CIDR notation.

**longer-prefixes**

Specifies all statistics for routes that match the specified route, or that have a longer prefix than the specified route.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**as-path-access-list**

Specifies an AS-path ACL.

**name**

Name of ACL.

**prefix-list**

Specifies an IP prefix list.

**name**

Name of prefix-list

## Modes

Privileged EXEC mode

Use this command with the **longer-prefixes** keyword to display routes that match a specified or longer IPv6 prefix. For example, if you specify 2001:db8::/16 longer-prefixes , then all routes with the prefix 2001:db8::/16 or that have a longer prefix (such as 2001:db8::/32) are displayed.

## Command Output

The `show ipv6 bgp filtered-routes detail` command displays the following information:

Output field	Description
Number of BGP4+ Routes matching display condition	The number of routes that matched the display parameters you entered. This is the number of routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route's status. The status code appears in the left column of the display, to the left of each route. The status codes are described in the command's output. The status column displays an "IF" for each filtered route.
Prefix	The network address and prefix.
Next Hop	The next-hop router for reaching the network from the device.
MED	The value of the route's MED attribute. If the route does not have a metric, this field is blank.
LocPrf	The degree of preference for this route relative to other routes in the local AS. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295.
Weight	The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight.
Status	<p>The route's status, which can be one or more of the following:</p> <ul style="list-style-type: none"> <li>• A - AGGREGATE - The route is an aggregate route for multiple networks.</li> <li>• B - BEST - BGP4+ has determined that this is the optimal route to the destination.</li> <li>• b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3 or static IPv6 routes).</li> <li>• C - CONFED_EBGP - The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation.</li> <li>• D - DAMPED - This route has been dampened (by the route dampening feature), and is currently unusable.</li> <li>• E - EBGP - The route was learned through a in another AS.</li> <li>• H - HISTORY - Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.</li> <li>• I - IBGP - The route was learned through a in the same AS.</li> <li>• L - LOCAL - The route originated on this device.</li> <li>• M - MULTIPATH - BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".</li> </ul> <p><b>NOTE</b> If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.</p> <ul style="list-style-type: none"> <li>• S - SUPPRESSED - This route was suppressed during aggregation and thus is not advertised to neighbors.</li> <li>• F - FILTERED - This route was filtered out by BGP4+ route policies on the device, but the device saved updates containing the filtered routes.</li> </ul>

## History

Release version	Command history
5.0.0	This command was introduced.



# show ipv6 bgp flap-statistics

Displays BGP4+ route-dampening statistics for all dampened routes with a variety of options.

## Syntax

```
show ipv6 bgp flap-statistics ipv6-addr mask [ longer-prefixes [ rbridge-id { rbridge-id | all } ] | rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp flap-statistics neighbor ipv6-addr [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp flap-statistics [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp flap-statistics regular-expression name [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of the destination network in dotted-decimal notation.

*mask*

IPv6 mask of the destination network in CIDR notation.

*longer-prefixes*

Displays statistics for routes that match the specified route or have a longer prefix than the specified route.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

**neighbor**

Displays statistics for routes learned from the specified neighbor.

*ipv6-addr*

IPv6 address of the neighbor.

**regular-expression**

Specifies routes matching the AS path regular expression.

*name*

Name of an AS-path filter or regular expression.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp flap-statistics** command displays the following information.

Output field	Description
Total number of flapping routes	The total number of routes in the device's BGP4+ route table that have changed state and thus have been marked as flapping routes.
Status code	Indicates the dampening status of the route, which can be one of the following: <ul style="list-style-type: none"> <li>&gt; - This is the best route among those in the BGP4+ route table to the route's destination.</li> <li>d - This route is currently dampened, and thus unusable.</li> <li>h - The route has a history of flapping and is unreachable now.</li> <li>* - The route has a history of flapping but is currently usable.</li> </ul>
Network	The destination network of the route.
From	The IPv6 address of the advertising peer.
Flaps	The number of flaps (state changes) the route has experienced.
Since	The amount of time (in hh:mm:ss) since the first flap of this route.
Reuse	The amount of time (in hh:mm:ss) after which the path is again available.
Path	The AS path of the route.

## Examples

This example shows sample output from the **show ipv6 bgp flap-statistics** command when no keyword is used.

```
device# show ipv6 bgp flap-statistics
```

```
Total number of flapping routes: 14
  Status Code >:best d:damped h:history *:valid
  Network      From      Flaps  Since  Reuse  Path
h> 2001:db8:2::/48 2001:db8:23::47 1    0 :0 :13 0 :0 :0 65001 4355 1 701
*> 2001:db8:34::/48 2001:db8:23::47 1    0 :1 :4  0 :0 :0 65001 4355 701 62
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ipv6 bgp neighbors

Displays configuration information and statistics for BGP4+ neighbors of the device.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr  
show ipv6 bgp neighbors last-packet-with-error [ rbridge-id { rbridge-id | all } | vrf vrf-name ]  
show ipv6 bgp neighbors routes-summary [ rbridge-id { rbridge-id | all } | vrf vrf-name ]  
show ipv6 bgp neighbors rbridge-id { rbridge-id | all }  
show ipv6 bgp neighbors vrf vrf-name
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**last-packet-with-error**

Displays information about the last packet from a neighbor that contained an error.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

**routes-summary**

Displays information about all route information received in UPDATE messages from BGP neighbors.

## Modes

Privileged EXEC mode

## Examples

The following is sample output from the **show ipv6 bgp neighbors** command when the **routes-summary** keyword is used.

```
device# # show ipv6 bgp neighbors routes-summary
  Total number of BGP Neighbors: 1
1  IP Address: 1000::1
Routes Accepted/Installed:0, Filtered/Kept:0, Filtered:0
  Routes Selected as BEST Routes:0
    BEST Routes not Installed in IP Forwarding Table:0
  Unreachable Routes (no IGP Route for NEXTHOP):0
  History Routes:0

NLRIs Received in Update Message:0, Withdraws:0 (0), Replacements:0
  NLRIs Discarded due to
    Maximum Prefix Limit:0, AS Loop:0
    Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
    Invalid Confed aspath:0, maxas-limit aspath:0
    Duplicated Originator_ID:0, Cluster_ID:0

Routes Advertised:0, To be Sent:9, To be Withdrawn:0
NLRIs Sent in Update Message:0, Withdraws:0, Replacements:0

Peer Out of Memory Count for:
  Receiving Update Messages:0, Accepting Routes (NLRI):0
  Attributes:0, Outbound Routes (RIB-out):0 Outbound Routes Holder:0
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.
7.0.0	This command was modified to include BGP add path configuration status.

# show ipv6 bgp neighbors advertised-routes

Displays the routes a BGP4+ networking device advertised to a neighbor during a BGP session.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr advertised-routes detail [ ipv6-addr mask [ rbridge-id { rbridge-id | all } ] | rbridge-id
{ rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr advertised-routes ipv6-addr mask [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr advertised-routes [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**detail**

Displays detailed information about routes advertised to a neighbor during a BGP session.

*ipv6-addr*

IPv6 address in dotted-decimal notation.

*mask*

IPv6 mask in CIDR notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp neighbors advertised-routes** command displays the following information.

Output field	Description
Number of BGP4+ Routes advertised to specified neighbor (appears only in display for all routes)	The number of routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output.

Output field	Description
Prefix	The advertised route's prefix.
Next Hop	The next-hop for reaching the advertised route from the device.
MED	The value of the advertised route's MED attribute. If the route does not have a metric, this field is blank.
LocPrf	The degree of preference for the advertised route relative to other routes in the local autonomous system. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference range is 0 - 4294967295.
Weight	The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight.
Status	The advertised route's status, which can be one or more of the following: <ul style="list-style-type: none"> <li>• A - AGGREGATE. The route is an aggregate route for multiple networks.</li> <li>• B - BEST. BGP4+ has determined that this is the optimal route to the destination.</li> <li>• b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3, or static IPv6 routes).</li> <li>• E - EBGP. The route was learned through a in another AS.</li> <li>• I - IBGP. The route was learned through a in the same AS.</li> <li>• L - LOCAL. The route originated on this device.</li> </ul>
AS-PATH	The AS-path information for the route.

The **show ipv6 bgp neighbors advertised-routes detail** command displays the following fields that are not described in the table above:

Output field	Description
Age	The age of the advertised route, in seconds.
Learned from Peer	The IPv6 address of the neighbor from which this route is learned. "Local Router" indicates that the device itself learned the route.
MED	The value of the advertised route's MED attribute. If the route does not have a metric, this field is blank.
Origin	The source of the route information. The origin can be one of the following: <ul style="list-style-type: none"> <li>• EGP - The routes with this set of attributes came to BGP4+ through EGP.</li> <li>• IGP - The routes with this set of attributes came to BGP4+ through IGP.</li> <li>• INCOMPLETE - The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPFv3.</li> </ul> <p>When BGP4+ compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.</p>
AS-PATH	The AS-path information for the route.

Output field	Description
Adj RIB out count	The number of routes in the device's current BGP4+ Routing Information Base (Adj-RIB-Out) for a specified neighbor.
Admin distance	The administrative distance of the route.

## Examples

This example shows sample output from the **show ipv6 bgp neighbors advertised-routes** command.

```
device# show ipv6 bgp neighbor 2001:54:54::54 advertised-routes

      There are 7 routes advertised to neighbor 2001:54:54::54
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
Prefix      Next Hop      MED      LocPrf    Weight  Status
1   fd80:122:122:122:101:101:0:122/128  2001:122:122::122
      0          100      101      BL
      AS_PATH:
2   fd80:122:122:122:103:103:0:122/128  2001:122:122::122
      0          100      103      BL
      AS_PATH:
3   fd80:122:122:122:105:105:0:122/128  2001:122:122::122
      0          100      105      BL
      AS_PATH:
4   131::1/128          2001:122:122::122
      1          100      32768   BL
      AS_PATH:
5   2001:122:131:125:131:1::/96  2001:3002::732
      1          100      0        BE
      AS_PATH: 65530
6   2001:abcd:1234:1234:1:2:1:0/112  2001:3002::733
      1          100      0        BE
      AS_PATH: 65530
7   2001:abcd:1234:1234:1:2:2:0/112  2001:3002::733
      1          100      0        BE
```

This example shows sample output from the **show ipv6 bgp neighbors advertised-routes** command when the **detail** keyword is used.

```
device# show ipv6 bgp neighbors 2001:54:54::54 advertised-routes detail
      There are 7 routes advertised to neighbor 2001:54:54::54
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
1   Prefix: fd80:122:122:122:101:101:0:122/128, Status: BL, Age: 4h23m34s
      NEXT_HOP: 2001:122:122::122, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 0, ORIGIN: igp, Weight: 101
      AS_PATH:
      Adj_RIB_out count: 20, Admin distance 0
2   Prefix: fd80:122:122:122:103:103:0:122/128, Status: BL, Age: 4h23m32s
      NEXT_HOP: 2001:122:122::122, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 0, ORIGIN: igp, Weight: 103
      AS_PATH:
      Adj_RIB_out count: 20, Admin distance 0
3   Prefix: fd80:122:122:122:105:105:0:122/128, Status: BL, Age: 4h23m31s
      NEXT_HOP: 2001:122:122::122, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 0, ORIGIN: igp, Weight: 105
      AS_PATH:
      Adj_RIB_out count: 20, Admin distance 0
4   Prefix: 131::1/128, Status: BL, Age: 4h23m49s
      NEXT_HOP: 2001:122:122::122, Learned from Peer: Local Router
      LOCAL_PREF: 100, MED: 1, ORIGIN: igp, Weight: 32768
      AS_PATH:
      Adj_RIB_out count: 20, Admin distance 1
5   Prefix: 2001:122:131:125:131:1::/96, Status: BE, Age: 2h39m44s
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.



# show ipv6 bgp neighbors flap-statistics

Displays configuration information and flap statistics for routes received from or sent to a neighbor.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr flap-statistics [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp neighbors flap-statistics** command displays the following information.

Output field	Description
Total number of flapping routes	The total number of routes in the neighbor's BGP4+ route table that have changed state and thus have been marked as flapping routes.
Status code	Indicates the status of the route, which can be one of the following: <ul style="list-style-type: none"> <li>&gt; - This is the best route among those in the neighbor's BGP4+ route table to the route's destination.</li> <li>d - This route is currently dampened, and thus unusable.</li> <li>h - The route has a history of flapping and is unreachable now.</li> <li>* - The route has a history of flapping but is currently usable.</li> </ul>
Network	The destination network of the route.
From	The IPv6 address of the advertising peer.
Flaps	The number of flaps (state changes) the route has experienced.
Since	The amount of time (in hh:mm:ss) since the first flap of this route.
Reuse	The amount of time (in hh:mm:ss) after which the path is again available.

Output field	Description
Path	The AS path of the route.

## Examples

The following is sample output from the **show ipv6 bgp neighbors flap-statistics** command.

```
device# show ipv6 bgp neighbors 2001:db8::110 flap-statistics

Total number of flapping routes: 14
  Status Code  >:best d:damped h:history *:valid
  Network      From      Flaps Since  Reuse      Path
h> 2001:db8:2::/48 10.90.213.77 1      0 :0 :13 0 :0 :0 65001 4355 1 701
*> 2001:db8:34::/48 10.90.213.77 1      0 :1 :4 0 :0 :0 65001 4355 701 62
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp neighbors last-packet-with-error

Displays information about the last packet that contained an error from any of a device's neighbors.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr last-packet-with-error [ decode ] [ rbridge-id { rbridge-id | all } ] [ vrf vrfname [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**decode**

Decodes last packet that contained an error from any of a device's neighbors.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp neighbors last-packet-with-error** command displays the following information.

Output field	Description
Total number of BGP Neighbors	The total number of configured neighbors for a device.
Last error	The error packet's contents decoded in a human-readable format or notification that no packets with an error were received.

## Examples

This example shows sample output from the **show ipv6 bgp neighbors last-packet-with-error** command when no packet from a specified neighbor contained an error.

```
device# show ipv6 bgp neighbors 1000::1 last-packet-with-error
No received packet with error logged for neighbor 1000::1
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp neighbors rbridge-id

Displays configuration information and statistics for BGP4+ neighbors by RBridge ID.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr rbridge-id { rbridge-id | all }
```

## Parameters

**all**

Specifies all RBridges.

*rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

This example shows sample output from the **show ipv6 bgp neighbors rbridge-id** command.

```
device# show ipv6 bgp neighbors 1000::1 rbridge-id 1
1  IP Address: 1000::1, AS: 100 (IBGP), RouterID: 10.0.0.0, VRF: default-vrf
   State: CONNECT, Time: 5d2h35m29s, KeepAliveTime: 60, HoldTime: 180
   Minimal Route Advertisement Interval: 0 seconds
   Address Family : IPV6 Unicast
   Route-map: (out) r1
   Messages:      Open      Update  KeepAlive  Notification  Refresh-Req
     Sent       : 0         0         0           0              0
     Received: 0         0         0           0              0
   Last Connection Reset Reason:Unknown
   Notification Sent:      Unspecified
   Notification Received: Unspecified
   Neighbor NLRI Negotiation:
     Peer configured for IPV6 unicast Routes
   Neighbor ipv6 MPLS Label Capability Negotiation:
   Neighbor AS4 Capability Negotiation:
   Outbound Policy Group:
     routemap: r1
     as-path-filter-list: myfilterlist
   ID: 3, Use Count: 1
   Last update time was 2042 sec ag
```

# show ipv6 bgp neighbors received

Displays Outbound Route Filters (ORFs) received from BGP4+ neighbors of the device.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr received
```

```
show ipv6 bgp neighbors ipv6-addr received detail [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr received prefix-filter [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr received rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**detail**

Displays detailed ORF information.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

**prefix-filter**

Displays the results for ORFs that are prefix-based.

## Modes

Privileged EXEC mode

## Examples

This example shows sample output from the **show ipv6 bgp neighbors received** command when the **prefix-filter** keyword is used.

```
device# show ipv6 bgp neighbors 4001::1 received prefix-filter

ip prefix-list: 2 entries
seq 1 permit 1001::/64
seq 2 permit 4001::/64
SW0)#show ipv6 bgp neighbors 4001::1 advertised-routes
There are 2 routes advertised to neighbor 4001::1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
Prefix Next Hop MED LocPrf Weight Status
1 1001::/64 4001::2 0 100 32768 BL
AS_PATH:
2 4001::/64 4001::2 0 100 32768 BL
AS_PATH:
Taurus
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ipv6 bgp neighbors received-routes

Lists all route information received in route updates from BGP4+ neighbors of the device since the soft-reconfiguration feature was enabled.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr received-routes detail [ rbridge-id { rbridge-id | all } ] vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr received-routes [ rbridge-id { rbridge-id | all } ] vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**detail**

Displays detailed route information.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp neighbors received-routes** command displays the following information.

Output field	Description
Number of BGP4+ Routes received from a neighbor	The number of routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output.
Prefix	The received route's prefix.
Next Hop	The IPv6 address of the next device that is used when forwarding a packet to the received route.
MED	The value of the route's MED attribute. If the route does not have a metric, this field is blank.
LocPrf	The degree of preference for the advertised route relative to other routes in the local autonomous system. When the BGP4+ algorithm



Output field	Description
	compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295.
Weight	The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight.
Status	<p>The advertised route's status, which can be one or more of the following:</p> <p>A - AGGREGATE. The route is an aggregate route for multiple networks.</p> <p>B - BEST. BGP4+ has determined that this is the optimal route to the destination.</p> <p>b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3, or static IPv6 routes).</p> <p>D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable.</p> <p>E - EBGP. The route was learned through a in another AS.</p> <p>H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.</p> <p>I - IBGP. The route was learned through a in the same autonomous system.</p> <p>L - LOCAL. The route originated on this device.</p> <p>M - MULTIPATH. BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".</p> <p><b>NOTE</b> If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.</p> <p>S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors.</p> <p>F - FILTERED. This route was filtered out by BGP4+ route policies on the device, but the saved updates containing the filtered routes.</p>

## Examples

This example shows sample output from the **show ipv6 bgp neighbors received-routes** command.

```
device# show ipv6 bgp neighbor 2001:db8::10 received-routes

There are 4 received routes from neighbor 2001:db8::10
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
  Prefix      Next Hop      Metric      LocPrf      Weight      Status
1   2001:db8:2002::/64  2001:db8::10    0    100    0    BE
AS_PATH: 400
2   2001:db8:2003::/64  2001:db8::10    1    100    0    BE
AS_PATH: 400
3   2001:db8:2004::/64  2001:db8::10    1    100    0    BE
AS_PATH: 400
4   2001:db8:2005::/64  2001:db8::10    1    100    0    BE
AS_PATH: 400
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ipv6 bgp neighbors rib-out-routes

Displays information about BGP4+ outbound RIB routes.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr rib-out-routes ipv6-addr mask [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr rib-out-routes detail ipv6-addr mask [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr rib-out-routes detail [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr rib-out-routes [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

**detail**

Displays detailed RIB route information.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp neighbors rib-out-routes** command displays the following information.

Output field	Description
Number of RIB_out routes for a specified neighbor (appears only in display for all RIB routes)	The number of RIB routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output.
Prefix	The RIB route's prefix.
Next Hop	The next-hop router for reaching the route from the device.
MED	The value of the advertised route's MED attribute. If the route does not have a metric, this field is blank.

Output field	Description
LocPrf	The degree of preference for the route relative to other routes in the local autonomous system. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295.
Weight	The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight.
Status	The RIB route's status, which can be one or more of the following: <ul style="list-style-type: none"> <li>A - AGGREGATE. The route is an aggregate route for multiple networks.</li> <li>B - BEST. BGP4+ has determined that this is the optimal route to the destination.</li> <li>E - EBGP. The route was learned through a in another autonomous system.</li> <li>I - IBGP. The route was learned through a in the same autonomous system.</li> <li>L - LOCAL. The route originated on this device.</li> </ul>
AS-PATH	The AS-path information for the route.

## Examples

This example shows sample output from the **show ipv6 bgp neighbors rib-out-routes** command.

```
device# show ipv6 bgp neighbors 2001:54:54::54 rib-out-routes

      There are 150 RIB_out routes for neighbor 2001:54:54::54
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
Prefix      Next Hop      MED      LocPrf      Weight      Status
1   fd80:122:122:122:101:101:0:122/128  ::      0      100      101      BL
      AS_PATH:
2   fd80:122:122:122:103:103:0:122/128  ::      0      100      103      BL
      AS_PATH:
3   fd80:122:122:122:105:105:0:122/128  ::      0      100      105      BL
      AS_PATH:
4   131::1/128      ::      1      100      32768     BL
      AS_PATH:
5   2001:122:131:125:131:1::/96  2001:3002::732
      AS_PATH: 65530
6   2001:abcd:1234:1234:1:2:1:0/112  2001:3002::733
      AS_PATH: 65530
7   2001:abcd:1234:1234:1:2:2:0/112  2001:3002::733
      AS_PATH: 65530
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ipv6 bgp neighbors routes

Lists a variety of route information received in UPDATE messages from BGP4+ neighbors.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr routes [ best | not-installed-best | unreachable [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ] ]
```

```
show ipv6 bgp neighbors ipv6-addr routes detail [ best | not-installed-best | unreachable [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ] ]
```

```
show ipv6 bgp neighbors ipv6-addr routes detail [ rbridge-id { rbridge-id | all } ] | vrf vrf-name ]
```

```
show ipv6 bgp neighbors ipv6-addr routes [ rbridge-id { rbridge-id | all } ] | vrf vrf-name
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**best**

Displays routes received from the neighbor that are the best BGP4+ routes to their destination.

**not-installed-best**

Displays routes received from the neighbor that are the best BGP4+ routes to their destination but were not installed in the route table because the device received better routes from other sources.

**unreachable**

Displays routes that are unreachable because the device does not have a valid OSPF or static route to the next hop.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

**detail**

Displays detailed information for the specified route types.

## Modes

Privileged EXEC mode

## Examples

This example shows sample output from the **show ipv6 bgp neighbors routes** command when the **best** keyword is used.

```
device# show ipv6 bgp neighbor 2001:db8::106 routes best

There are 2 accepted routes from neighbor 2001:db8::106
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
Prefix Next Hop MED LocPrf Weight Status
1 2001:db8:2002::/48 2001:db8::106 1 100 0 BE
AS_PATH: 65001
2 2001:db8:2002:1234::/64 2001:db8::106 1 100 0 BE
AS_PATH: 65001
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.

# show ipv6 bgp neighbors routes-summary

Lists all route information received in UPDATE messages from BGP4+ neighbors.

## Syntax

```
show ipv6 bgp neighbors ipv6-addr routes-summary [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a specified route in dotted-decimal notation.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp neighbors routes-summary** command displays the following information.

Output field	Description
IP Address	The IPv6 address of the neighbor
Routes Received	How many routes the device has received from the neighbor during the current BGP4+ session: <ul style="list-style-type: none"> <li>Accepted or Installed - Indicates how many of the received routes the device accepted and installed in the BGP4+ route table.</li> <li>Filtered or Kept - Indicates how many routes were filtered out, but were nonetheless retained in memory for use by the soft reconfiguration feature.</li> <li>Filtered - Indicates how many of the received routes were filtered out.</li> </ul>
Routes Selected as BEST Routes	The number of routes that the device selected as the best routes to their destinations.
BEST Routes not Installed in IPv6 Forwarding Table	The number of routes received from the neighbor that are the best BGP4+ routes to their destinations, but were nonetheless not installed in the IPv6 route table because the device received better routes from other sources (such as OSPFv3, or static IPv6 routes).

Output field	Description
Unreachable Routes	The number of routes received from the neighbor that are unreachable because the device does not have a valid OSPFv3, or static IPv6 route to the next hop.
History Routes	The number of routes that are down but are being retained for route flap dampening purposes.
NLRIs Received in Update Message	The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages: <ul style="list-style-type: none"> <li>• Withdraws - The number of withdrawn routes the device has received.</li> <li>• Replacements - The number of replacement routes the device has received.</li> </ul>
NLRIs Discarded due to	Indicates the number of times the device discarded an NLRI for the neighbor due to the following reasons: <ul style="list-style-type: none"> <li>• Maximum Prefix Limit - The device's configured maximum prefix amount had been reached.</li> <li>• AS Loop - An AS loop occurred. An AS loop occurs when the BGP4+ AS-path attribute contains the local AS number.</li> <li>• Invalid Nexthop Address - The next hop value was not acceptable.</li> <li>• Duplicated Originator_ID - The originator ID was the same as the local router ID.</li> <li>• Cluster_ID - The cluster list contained the local cluster ID, or contained the local router ID (see above) if the cluster ID is not configured.</li> </ul>
Routes Advertised	The number of routes the device has advertised to this neighbor: <ul style="list-style-type: none"> <li>• To be Sent - The number of routes the device has queued to send to this neighbor.</li> <li>• To be Withdrawn - The number of NLRIs for withdrawing routes the device has queued up to send to this neighbor in UPDATE messages.</li> </ul>
NLRIs Sent in Update Message	The number of NLRIs for new routes the device has sent to this neighbor in UPDATE messages: <ul style="list-style-type: none"> <li>• Withdraws - The number of routes the device has sent to the neighbor to withdraw.</li> <li>• Replacements - The number of routes the device has sent to the neighbor to replace routes the neighbor already has.</li> </ul>
Peer Out of Memory Count for	Statistics for the times the device has run out of BGP4+ memory for the neighbor during the current BGP4+ session: <ul style="list-style-type: none"> <li>• Receiving Update Messages - The number of times UPDATE messages were discarded because there was no memory for attribute entries.</li> <li>• Accepting Routes(NLRI) - The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count.</li> <li>• Attributes - The number of times there was no memory for BGP4+ attribute entries.</li> <li>• Outbound Routes (RIB-out) - The number of times there was no memory to place a "best" route into the neighbor's route information base (Adj-RIB-Out) for routes to be advertised.</li> <li>• Outbound Routes Holder - For debugging purposes only.</li> </ul>



## Examples

This example shows sample output from the **show ipv6 bgp neighbors routes-summary** command.

```
device# show ipv6 bgp neighbors routes-summary

Total number of BGP Neighbors: 1
1 IP Address: 5001::1
Routes Accepted/Installed:3, Filtered/Kept:0, Filtered:0
Routes Selected as BEST Routes:3
BEST Routes not Installed in IP Forwarding Table:0
Unreachable Routes (no IGP Route for NEXTHOP):0
History Routes:0
NLRIs Received in Update Message:3, Withdraws:0 (0), Replacements:0
NLRIs Discarded due to
Maximum Prefix Limit:0, AS Loop:0
Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
Invalid Confed aspath:0, maxas-limit aspath:0
Duplicated Originator_ID:0, Cluster_ID:0
Routes Advertised:0, To be Sent:0, To be Withdrawn:0
NLRIs Sent in Update Message:0, Withdraws:0, Replacements:0
Peer Out of Memory Count for:
Receiving Update Messages:0, Accepting Routes(NLRI):0
Attributes:0, Outbound Routes(RIB-out):0 Outbound Routes Holder:0
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp peer-group

Displays peer-group information.

## Syntax

```
show ipv6 bgp peer-group [ peer-group-name [ rbridge-id { rbridge-id | all } ] ]
```

```
show ipv6 bgp peer-group [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

*peer-group-name*

Peer-group name configured by the **neighbor** *peer-group-name* command.

**rbridge-id**

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

The following is sample output from the **show ipv6 bgp peer-group** command.

```
device# # show ipv6 bgp peer-group
do show ipv6 bgp peer-group
1 BGP peer-group is P1, Remote AS: 1
Address family : IPV4 Unicast
activate
Address family : IPV4 Multicast
no activate
Address family : IPV6 Unicast
activate
Address family : IPV6 Multicast
no activate
Address family : VENV4 Unicast
no activate
Address family : L2VPN VPLS
no activate
Members:
IP Address: 2001::1
IP Address: 2001:0:0:1::1
IP Address: 10.1.0.1
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp rbridge-id

Displays BGP4+ route information by RBridge ID.

## Syntax

```
show ipv6 bgp rbridge-id { rbridge-id | all } [ vrf vrf-name ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridges.

vrf *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example displays route information for RBridge ID 5 and VRF instance "red".

```
device# show ipv6 bgp rbridge-id 5 vrf red
```

## History

Release version	Command history
6.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp routes

Displays BGP4+ route information that can be filtered using various options.

## Syntax

```
show ipv6 bgp routes [ num | ipv6-address/prefix | age num | as-path-access-list name | best | cidr-only | community-
access-list name | community-reg-expression expression | detail | local | neighbor ipv6-addr | nexthop ipv6-addr | no-
best | not-installed-best | prefix-list string | regular-expression name | route-map name | summary | unreachable ]
[ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

*num*

Table entry at which the display starts.

*ipv6-address/prefix*

Table entry at which the display starts.

**age**

Displays BGP4+ route information that is filtered by age.

**as-path-access-list**

Displays BGP4+ route information that is filtered by autonomous system (AS)-path access control list (ACL).

**best**

Displays BGP4+ route information that the device selected as best routes.

**cidr-only**

Displays BGP4+ routes whose network masks do not match their class network length.

**community-access-list** *name*

Displays BGP4+ route information for an AS-path community access list.

**community-reg-expression** *expression*

Displays BGP4+ route information for an ordered community-list regular expression.

**detail**

Displays BGP4+ detailed route information.

**local**

Displays BGP4+ route information about selected local routes.

**neighbor** *ipv6-addr*

Displays BGP4+ route information about selected BGP neighbors.

**nexthop** *ipv6-addr*

Displays BGP4+ route information about routes that are received from the specified next hop.

**no-best**

Displays BGP4+ route information that the device selected as not best routes.

**not-installed-best**

Displays BGP4+ route information about best routes that are not installed.

**prefix-list** *string*

Displays BGP4+ route information that is filtered by prefix list.

**regular-expression** *name*

Displays BGP4+ route information about routes that are associated with the specified regular expression.

**route-map** *name*

Displays BGP4+ route information about routes that use the specified route map.

**summary**

Displays BGP4+ summary route information.

**unreachable**

Displays BGP4+ route information about routes whose destinations are unreachable through any of the BGP4 paths in the BGP4 route table.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp routes** command displays the following information.

Output field	Description
Number of BGP4+ Routes	The number of routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output.
Prefix	The route's prefix.
Next Hop	For normal IPv6 routes, next hop is the next hop IPv6 router to reach the destination. For the 6PE routes, next hop is the IPv4-mapped IPv6 address of the peer 6PE router.
MED	The value of the route's MED attribute. If the route does not have a metric, this field is blank.
LocPrf	The degree of preference for the advertised route relative to other routes in the local AS. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295.
Weight	The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight.

Output field	Description
Status	<p>The route's status, which can be one or more of the following:</p> <ul style="list-style-type: none"> <li>A - AGGREGATE. The route is an aggregate route for multiple networks.</li> <li>B - BEST. BGP4+ has determined that this is the optimal route to the destination.</li> <li>b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3 or static IPv6 routes).</li> <li>C - CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation.</li> <li>D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable.</li> <li>E - EBGP. The route was learned through a in another AS.</li> <li>H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.</li> <li>I - IBGP. The route was learned through a in the same AS.</li> <li>L - LOCAL. The route originated on this.</li> <li>M - MULTIPATH. BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".</li> </ul> <p><b>NOTE</b> If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.</p> <ul style="list-style-type: none"> <li>S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors.</li> </ul>
AS-PATH	The AS-path information for the route.

## Examples

The following is sample output from the **show ipv6 bgp routes** command.

```
device# show ipv6 bgp routes

Total number of BGP Routes: 6
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix          Next Hop          MED          LocPrf      Weight Status
1      57:7000:3:22:abc:1::/128  2001:700:122:57::57
          AS_PATH: 7000 322
          100          0          BE
2      57:7000:3:22:abc:1:0:2/128  2001:700:122:57::57
          AS_PATH: 7000 322
          100          0          BE
3      57:7000:3:22:abc:1:0:4/128  2001:700:122:57::57
          AS_PATH: 7000 322
          100          0          BE
4      57:7000:3:22:abc:1:0:6/128  2001:700:122:57::57
          AS_PATH: 7000 322
          100          0          BE
5      57:7000:3:22:abc:1:0:8/128  2001:700:122:57::57
          AS_PATH: 7000 322
          100          0          BE
6      57:7000:3:22:abc:1:0:a/128  2001:700:122:57::57
          AS_PATH: 7000 322
          100          0          BE
```

The following is sample output from the **show ipv6 bgp routes** command using the **summary** keyword.

```
device# show ipv6 bgp routes summary

Total number of BGP routes (NLRIs) Installed      : 558
Distinct BGP destination networks                : 428
Filtered bgp routes for soft reconfig            : 0
Routes originated by this router                  : 19
Routes selected as BEST routes                   : 417
BEST routes not installed in IP forwarding table  : 0
Unreachable routes (no IGP route for NEXTHOP)   : 22
IBGP routes selected as best routes              : 102
EBGP routes selected as best routes              : 296
```

The following is sample output from the **show ipv6 bgp routes** command using the **local** keyword.

```
device# show ipv6 bgp routes local
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
Prefix      Next Hop      MED      LocPrf      Weight Status
1  131::1/128      ::          1         100         32768  BL
   AS_PATH:
2  2001::107:6133:2007:1::/112  2001:2007::201
   107         100         32768  BL
   AS_PATH:
3  2001::107:6133:2007:2::/112  2001:2007::202
   107         100         32768  BL
   AS_PATH:
4  2001::107:6133:2007:3::/112  2001:2007::203
   107         100         32768  BL
   AS_PATH:
5  2001::107:6133:2007:4::/112  2001:2007::204
   107         100         32768  BL
   AS_PATH:
6  2001::107:6133:2007:5::/112  2001:2007::205
   107         100         32768  BL
   AS_PATH:
7  2001::107:6133:2007:6::/112  2001:2007::206
   107         100         32768  BL
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf vrf-name</b> parameter was added to support Multi-VRF.
7.0.0	Command output was modified to include details about BGP additional paths.



# show ipv6 bgp routes community

Displays BGP4+ route information that is filtered by community and other options.

## Syntax

```
show ip bgp routes [ community num | internet | local-as | no-advertise | no-export ] [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

### community

Displays routes filtered by a variety of communities.

*num*

Specific community member.

### internet

Displays routes for the Internet community.

### local-as

Displays routes for a local sub-AS within the confederation.

### no-advertise

Displays routes with this community that cannot be advertised to any other BGP4+ devices at all.

### no-export

Displays routes for the community of sub-ASs within a confederation.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

### vrf *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Examples

This example shows output from the **show ipv6 bgp routes community** command when the **internet** keyword is used.

```

device# show ipv6 bgp routes community internet
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
Prefix          Next Hop      MED      LocPrf    Weight  Status
1  2001:131:1:1::131/128  1::1
      AS_PATH: 33091
2  2002:fd0:131:1234:1::/80  1::1
      AS_PATH: 33091
3  fd01:131:122:1::1/128  1::1
      AS_PATH: 33091
4  fd31:131:122:125::1/128  1::1
      AS_PATH: 33091
5  fd31:131:131:131::/64  1::1
      AS_PATH: 33091
6  fd31:131:131:131:1::/80  1::1
      AS_PATH: 33091 4294967295
7  fd31:131:131:131:2::/80  1::1
      AS_PATH: 33091
8  fd80:131:122:125:131:ff:1:0/112  1::1
      AS_PATH: 33091

```

## History

Release version	Command history
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 bgp summary

Displays summarized information about the status of all BGP connections.

## Syntax

```
show ipv6 bgp summary [ rbridge-id { rbridge-id | all } | vrf vrf-name ]
```

## Parameters

### rbridge-id

Specifies a RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

### vrf vrf-name

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 bgp summary** command displays the following information.

Output field	Description
Router ID	The device's router ID.
Local AS Number	The BGP4+ AS number in which the device resides.
Confederation Identifier	The autonomous system number of the confederation in which the device resides.
Confederation Peers	The numbers of the local autonomous systems contained in the confederation. This list matches the confederation peer list you configure on the device.
Maximum Number of Paths Supported for Load Sharing	The maximum number of route paths across which the device can balance traffic to the same destination. The feature is enabled by default but the default number of paths is 1. You can increase the number from 2 - 8 paths.
Number of Neighbors Configured	The number of BGP4+ neighbors configured on this device.
Number of Routes Installed	The number of BGP4+ routes in the device's BGP4+ route table.
Number of Routes Advertising to All Neighbors	The total of the RtSent and RTToSend columns for all neighbors.
Number of Attribute Entries Installed	The number of BGP4+ route-attribute entries in the route-attributes table.
Neighbor Address	The IPv6 addresses of this BGP4+ neighbors.
AS#	The autonomous system number.

Output field	Description
State	<p>The state of this neighbor session with each neighbor. The states are from this perspective of the session, not the neighbor's perspective. The state values can be one of the following for each:</p> <ul style="list-style-type: none"> <li>• IDLE - The BGP4+ process is waiting to be started. Usually, enabling BGP4+ or establishing a neighbor session starts the BGP4+ process. <ul style="list-style-type: none"> <li>- A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.</li> </ul> </li> <li>• ADMND - The neighbor has been administratively shut down. <ul style="list-style-type: none"> <li>- A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.</li> </ul> </li> <li>• CONNECT - BGP4+ is waiting for the connection process for the TCP neighbor session to be completed.</li> <li>• ACTIVE - BGP4+ is waiting for a TCP connection from the neighbor.</li> </ul> <p><b>NOTE</b> If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection.</p> <ul style="list-style-type: none"> <li>• OPEN SENT - BGP4+ is waiting for an Open message from the neighbor.</li> <li>• OPEN CONFIRM - BGP4+ has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle.</li> <li>• ESTABLISHED - BGP4+ is ready to exchange UPDATE packets with the neighbor. <ul style="list-style-type: none"> <li>- If there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed.</li> </ul> </li> </ul> <p><b>NOTE</b> If you display information for the neighbor using the <b>show ipv6 bgp neighbor</b> command, the TCP receiver queue value will be greater than 0.</p>
Time	The time that has passed since the state last changed.
Accepted	The number of routes received from the neighbor that this installed in the BGP4+ route table. Usually, this number is lower than the RoutesRcvd number. The difference indicates that this filtered out some of the routes received in the UPDATE messages.
Filtered	<p>The routes or prefixes that have been filtered out.</p> <ul style="list-style-type: none"> <li>• If soft reconfiguration is enabled, this field shows how many routes were filtered out (not placed in the BGP4+ route table) but retained in memory.</li> <li>• If soft reconfiguration is not enabled, this field shows the number of BGP4+ routes that have been filtered out.</li> </ul>
Sent	The number of BGP4+ routes sent to the neighbor.
ToSend	<p>The number of routes queued to send to this neighbor.</p> <ul style="list-style-type: none"> <li>• s - GR neighbor is in a stale state</li> </ul>

Output field	Description
	<ul style="list-style-type: none"> <li>r - GR neighbor is in a restarting state</li> <li>&lt; - GR neighbor is waiting for an EOR</li> </ul>

## Examples

The following is sample output from the **show ipv6 bgp summary** command.

```
device# show ipv6 bgp summary
```

```
BGP4 Summary
Router ID: 122.122.122.122   Local AS Number: 122
Confederation Identifier: not configured
Confederation Peers:
Cluster ID: 122
Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
Number of Neighbors Configured: 20, UP: 15
Number of Routes Installed: 219, Uses 20805 bytes
Number of Routes Advertising to All Neighbors: 2802 (440 entries), Uses 26400 bytes
Number of Attribute Entries Installed: 31, Uses 2852 bytes
Neighbor Address  AS#      State   Time      Rt:Accepted  Filtered  Sent      ToSend
2001:54:54::54   122      ESTAB  0h19m58s  0            0         146      0
2001:55:55::55   122      ESTAB  0h19m54s  1            0         146      0
2001:122:53::53  6000     ESTAB  0h22m39s  50           0         147      0
2001:122:534:2::534
                    534      ESTAB  0h 3m20s  10           0         137      0
2001:125:125::125 122      CONN   0h11m33s  0            0         0        -
```

The following is sample output from the **show ipv6 bgp summary** command when a GR neighbor is in a stale state.

```
device# show ipv6 bgp summary
```

```
BGP4 Summary
Router ID: 140.1.1.3   Local AS Number: 50
Confederation Identifier: not configured
Confederation Peers:
Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
Number of Neighbors Configured: 4, UP: 1
Number of Routes Installed: 13, Uses 1235 bytes
Number of Routes Advertising to All Neighbors: 32 (16 entries), Uses 960 bytes
Number of Attribute Entries Installed: 3, Uses 276 bytes
Neighbor Address  AS#      State   Time      Rt:Accepted  Filtered  Sent      ToSend
10.10.10.1       50      ACTIVS  0h 0m 0s   10           0         3         0 <
```

The following is sample output from the **show ipv6 bgp summary** command when a GR neighbor is in a restarting state.

```
device# show ipv6 bgp summary
```

```
BGP4 Summary
Router ID: 140.1.1.3   Local AS Number: 50
Confederation Identifier: not configured
Confederation Peers:
Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
Number of Neighbors Configured: 4, UP: 1
Number of Routes Installed: 13, Uses 1235 bytes
Number of Routes Advertising to All Neighbors: 32 (16 entries), Uses 960 bytes
Number of Attribute Entries Installed: 3, Uses 276 bytes
Neighbor Address  AS#      State   Time      Rt:Accepted  Filtered  Sent      ToSend
10.10.10.1       50      OPENSr  0h 0m 2s   10           0         0         3 <
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	The <b>vrf</b> <i>vrf-name</i> parameter was added to support Multi-VRF.

# show ipv6 counters interface

Displays the counters on an IPv6 interface.

## Syntax

```
show ipv6 counters interface { <N>gigabitethernet rbridge-id / slot / port | loopback port_number | port-channel number | ve
vlan_id }
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### loopback port\_number

Specifies the port number for the loopback interface. The range is from 1 through 255.

### port-channel number

Specifies a port-channel interface. Range is from 1 through 6144.

### ve vlan\_id

Specifies the VLAN ID of a virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Examples

The following example shows counters on an Ethernet interface.

```
device# show ipv6 counters interface te 1/0/10
Interface TenGigabitEthernet 1/0/10 IPv6 statistics (ifindex 402980872)
  Ip6OutRequests                21
  Ip6OutMcastPkts              30
  Ip6OutOctets                  2024
  Ip6OutMcastOctets            3008
  Icmp6OutMsgs                  21
  Icmp6OutRouterAdvertisements  6
  Icmp6OutNeighborSolicits     6
  Icmp6OutMLDv2Reports         9
  Icmp6OutType134               6
  Icmp6OutType135               6
  Icmp6OutType143               25
```

show ipv6 counters interface

The following example shows counters on a port-channel.

```
device# show ipv6 counters interface port-channel 10
Interface Port-channel 10 IPv6 statistics (ifindex 671088650)
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.



# show ipv6 dhcp relay address interface

Displays IPv6 DHCP Relay addresses configured on supported interfaces.

## Syntax

```
show ipv6 dhcp relay address interface <N>gigabitethernet rbridge-id / slot / port
```

```
show ipv6 dhcp relay address interface port-channel number [ rbridge-id { rbridge-id | range | all } ]
```

```
show ipv6 dhcp relay address interface ve vlan-id [ rbridge-id { rbridge-id | range | all } ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

### ve

VE interface.

### *vlan-id*

VLAN identification for interface.

### rbridge-id

Specifies an RBridge, multiple RBridges, or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

*range*

Specifies multiple RBridge IDs. You can specify a range (for example, 3-5), a comma-separated list (for example, 1,3,5,6), or you can combine a range with a list (for example, 1-5,6,8). In a range string, no spaces are allowed.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays configured IPv6 DHCP Relay addresses on a specified physical interface.

```
device# show ipv6 dhcp relay address interface tengigabitethernet 1/0/24
                                     Rbridge Id: 3
-----
Interface      Relay Address      VRF Name      Outgoing Interface
-----
Te 3/0/21      4001::10           default-vrf
Te 3/0/21      fe80::8            blue          ve 100
```

The following example displays configured IPv6 DHCP Relay addresses on a specified port channel.

```
device# show ipv6 dhcp relay address interface port-channel 10
                                     Rbridge Id: 2
-----
DHCPv6 unique identifier(DUID): 0102044a0027f8cad50e
-----
Interface      Relay Address      VRF Name      Outgoing Interface
-----
Po 10          2::2              default-vrf
```

The following example displays configured IPv6 DHCP Relay addresses on a specified VE interface and RBridge.

```
device# show ipv6 dhcp relay address int ve 300 rbridge-id 1
                                     Rbridge Id: 1
-----
Interface      Relay Address      VRF Name
-----
Ve 300         5001:1234:1234:2101:1234:1234:3103:1234  default-vrf
```

The following example displays configured IPv6 DHCP Relay addresses on a specified VE interface and RBridges.

```
device# show ipv6 dhcp relay address interface ve 300 rbridge-id 1,3
                                     Rbridge Id: 1
-----
Interface      Relay Address      VRF Name
-----
Ve 300         5001:1234:1234:2101:1234:1234:3103:1234  default-vrf

                                     Rbridge Id: 3
-----
Ve 300         2::2              default-vrf
```

## History

Release version	Command history
5.0.1	This command was introduced.
7.0.0	This command was modified to support port-channels.

# show ipv6 dhcp relay address rbridge-id

Displays IPv6 DHCP Relay addresses.

## Syntax

```
show ipv6 dhcp relay address rbridge-id rbridge-id | all | range
```

## Command Default

If the *rbridge-id* parameter is omitted, IP DHCP Relay addresses display for the local switch.

## Parameters

*rbridge-id*

Specifies an RBridge. You can specify multiple RBridge IDs, separated by commas.

all

Specifies all RBridges.

range

A range of RBridge IDs separated by a dashes or commas, or both. Range can be discontinuous, for example, 1-3,5. Spaces are not allowed.

For example: 1-3 - RBridge ID 1 through 3 1-3, 5 - RBridge ID 1 through 3 and RBridge ID 5 1, 3, 5, 6 - RBridge ID 1, 3, 5, and 6. in the range string

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays the IPv6 address and Virtual Routing and Forwarding (VRF) name for all interfaces with configured IPv6 DHCP Relay addresses on a local switch, specific switches, or all switches in a cluster. No spaces are allowed in the *range* string. The range does not need to be contiguous (for example, 1-2,5).

## Examples

The following example displays addresses configured on a specified RBridge ID.

```
device# show ipv6 dhcp relay address
                               Rbridge Id:    3
                               -----
Interface                       Relay Address                               VRF Name           Outgoing Interface
-----
Te 3/0/21                       4001::101                               default-vrf
Te 3/0/21                       fe80::8                                  blue                ve 100
Ve 200                          5001:1234:1234:2101:1234:1234:3103:1234  default-vrf        Te 3/0/3
```

The following example displays addresses configured on all switches in a cluster.

```
device# show ipv6 dhcp rel address rbridge-id all
                                     Rbridge Id: 1
                                     -----
Interface                            Relay Address                            VRF Name
-----                            -----                            -----
Te 1/0/24                            2.3.4.5                            default-vrf
Ve 300                                10.0.1.2                            default-vrf
                                     Rbridge Id: 3
                                     -----
Interface                            Relay Address                            VRF Name
-----                            -----                            -----
Ve 300                                10.0.0.5                            default-vrf
```

## History

Release version	Command history
5.0.1	This command was introduced.

# show ipv6 dhcp relay statistics

Displays general information about the DHCPv6 Relay function.

## Syntax

```
show ipv6 dhcp relay statistics rbridge-id rbridge-id | all | range ]
```

## Command Default

If the **rbridge-id** parameter is omitted, IPv6 DHCP Relay statistics display for the local switch.

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge. You can specify multiple RBridge IDs, separated by commas.

**all**

Specifies all RBridges.

*range*

A range of RBridge IDs separated by a dashes or commas, for example:

1-3 - RBridge ID 1 through 3  
 1-3, 5 - RBridge ID 1 through 3 and RBridge ID 5  
 1, 3, 5, 6 - RBridge ID 1, 3, 5, and 6

## Modes

Privileged EXEC mode

## Usage Guidelines

No spaces are allowed in the *range* string. The range does not need to be contiguous (for example, 1-2,5). You can also specify **all** for all RBridge IDs in a cluster. To display addresses for configured interfaces on a local switch, an RBridge ID parameter is not required.

The **show ipv6 dhcp relay statistics** command displays the following information about the IP DHCP Relay function for IP DHCP Relay addresses configured on a local switch, specific switches, or all switches in a cluster:

- Number of DHCP Error packets dropped.
- Number of DHCP SOLICIT, REQUEST, CONFIRM, RENEW, REBIND, RELEASE, DECLINE, INFORMATION-REQUEST, RELAY-FORWARD, RELAY-REPLY packets received.
- Number of DHCP RELAY-FORWARD, REPLY packets sent.

## Examples

The following example displays statistics for specific RBridge IDs.

```
device# show ipv6 dhcp relay statistics rbridge-id 6

Rbridge Id:    6
Packets dropped      : 0
  Error            : 0
Packets received    : 6
  SOLICIT          : 2
  REQUEST          : 1
  CONFIRM          : 0
  RENEW            : 0
  REBIND           : 0
  RELEASE          : 0
  DECLINE          : 0
  INFORMATION-REQUEST : 0
  RELAY-FORWARD    : 0
  RELAY-REPLY      : 3
Packets sent        : 6
  RELAY-FORWARD    : 3
  REPLY            : 3
```

## History

Release version	Command history
5.0.1	This command was introduced.

# show ipv6 fabric-virtual-gateway

Displays IPv6 Fabric-Virtual-Gateway session details.

## Syntax

```
show ipv6 fabric-virtual-gateway { detail | summary | interface ve vlan-id } [ rbridge-id { rbridge-id | all }
```

## Parameters

### detail

Lists the IPv6 Fabric-Virtual-Gateway configuration in detail.

### summary

Lists a summary of the IPv6 Fabric-Virtual-Gateway configuration.

### interface ve *vlan-id*

Displays IPv6 Fabric-Virtual-Gateway configuration for the specified VE interface.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Examples

The following example displays IPv6 Fabric-Virtual-Gateway session details.

```
device# show ipv6 fabric-virtual-gateway detail

=====Rbridge-id:51=====
Total number of IPv6 Fabric Virtual Gateway sessions   : 1
Total number of sessions in Active state              : 1
Total number of sessions in InActive state            : 0
Total number of sessions in Init state                : 0

Interface: Ve 213; Ifindex: 1207959765
Admin Status: Enabled
Description :
Address family: IPV6      State: Active
ARP responder Rbridge-id: 51
Gateway IP: 44:44::1/64
Gateway MAC Address: 02e0.5200.02fe
Load balancing configuration: Enabled
Load balancing current status: Enabled
Load balancing threshold priority: unset
Gratuitous ARP Timer: Disabled
Hold time: 0 sec (default: 0 sec)
  Total no. of state changes: 1
  ND Advertisements Sent: 1
Last state change: 1d.11h.21m.7s ago
Track Priority: 0
```

The following example displays IPv6 Fabric-Virtual-Gateway session details for the given VE interface.

```
device# show ipv6 fabric-virtual-gateway interface ve 2004

Please replace the output with the following
sw0# show ip fabric-virtual-gateway int ve 2004

=====Rbridge-id:51=====
Interface  Admin   State      Gateway      ARP          Load          Threshold  Track
          State            IP Address  Responder    Balancing     Priority      Priority
=====
Ve 2004   Enabled Active     44:44::1/64 Rbr-id 52    Enabled       100         0
```

## History

Release version	Command history
5.0.1	This command was introduced.



# show ipv6 interface

Displays details of IPv6 interfaces.

## Syntax

```
show ipv6 interface { brief | loopback number | ve vlan_id } [ rbridge-id { all | rbridge-id ]
```

```
show ipv6 interface { <N>gigabitethernet [ rbridge-id / ] slot / port | port-channel number }
```

## Parameters

### brief

Specifies brief interface information.

### loopback *number*

Specifies a loopback port number. The range is from 1 through 255.

### ve *vlan\_id*

Specifies the VLAN ID of a virtual Ethernet (VE) interface.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

### <N> gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, tengigabitethernet specifies a 10-Gb Ethernet port). The use of gigabitethernet without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

(Optional) Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

## Modes

Privileged EXEC mode

## Examples

The following example illustrates the output of the **show ipv6 interface** command with an Ethernet interface specified.

```
device# show ipv6 interface tengigabitethernet 1/0/10
TenGigabitEthernet 1/0/10 is up protocol is up
IPv6 Address: 1111::2222/64 Primary Confirmed
IPv6 Address: fe80::227:f8ff:fe88:e4df/128 Link Local Confirmed
IPv6 MTU: 1500
Vrf : default-vrf
```

The following example illustrates the output of the **show ipv6 interface** command with a virtual Ethernet and RBridge ID specified.

```
device# show ipv6 interface ve 10 rbridge 1

Ve 10 is administratively down protocol is down
IPv6 Address: fe80::/128 Link Local Wait Confirm
IPv6 MTU: 1500
Vrf : default-vrf
```

The following example illustrates the output of the **show ipv6 interface** command with a virtual Ethernet of 1 specified.

```
device# show ipv6 interface ve 1
Ve 1 is up protocol is up
IPv6 Address: 1001::4/64 Primary Confirmed
IPv6 Address: fe80::205:33ff:fee4:4011/128 Link Local Confirmed
IPv6 Address: fe80::205:33ff:fee5:ba86/128 Link Local Confirmed
IPv6 multicast groups locally joined:
  ff02::1
  ff02::2    ff02::1:ff00:4    ff02::1:ffe4:4011
  ff02::1:ffe5:ba86
IPv6 MTU: 1500
Vrf : default-vrf
```

The following example illustrates the output of the **show ipv6 interface** command applied directly on an interface with the **brief** option.

```
device(conf-if-te-1/0/10)# do show ipv6 int br
Interface      Vrf              Status           Protocol         IPv6-Address
=====
FortyGigabitEthernet 1/0/49          default-vrf      up               up               unassigned
FortyGigabitEthernet 1/0/50          default-vrf      up               down             unassigned
FortyGigabitEthernet 1/0/51          default-vrf      up               down             unassigned
FortyGigabitEthernet 1/0/52          default-vrf      up               down             unassigned
TenGigabitEthernet 1/0/1           default-vrf      up               down             unassigned
TenGigabitEthernet 1/0/2           default-vrf      up               down             unassigned
TenGigabitEthernet 1/0/10          default-vrf      up               up               1111::2222/64
```

The following example illustrates the output of the **show ipv6 interface** command with the **brief** option specified. Note that interface 3/2/1 is down due to detection of repeated MAC-moves.

```
device# show ipv6 interface brief
```

Interface	Vrf	Status	Protocol	IPv6-Address
TenGigabitEthernet 3/2/1 unassigned detection)	default-vrf	admin down	down (Repeated Mac-move	
TenGigabitEthernet 3/2/2 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/3 unassigned	default-vrf	up	up	
TenGigabitEthernet 3/2/4 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/5 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/6 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/7 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/8 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/9 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/10 unassigned	default-vrf	up	up	
TenGigabitEthernet 3/2/11 unassigned	default-vrf	up	up	
TenGigabitEthernet 3/2/12 unassigned	default-vrf	up	up	
TenGigabitEthernet 3/2/13 unassigned	default-vrf	up	up	
TenGigabitEthernet 3/2/14 unassigned	default-vrf	up	down	
TenGigabitEthernet 3/2/15	default-vrf	up	down	

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

## Related Commands

[show mac-address-table mac-move](#)

# show ipv6 mld groups

Displays information about a specific IPv6 MLDv1 group or a VLAN.

## Syntax

```
show ipv6 mld groups [ ipv6address ] [ interface vlan vlan_id | summary ]
```

## Parameters

*ipv6address*

A multicast group address.

**interface vlan**

Specifies a VLAN ID.

*vlan\_id*

A VLAN ID. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

**summary**

Displays summary information.

## Modes

Privileged EXEC mode

## Examples

To display information about all IPv6 MLDv1 groups:

```
device# show ipv6 mld groups
```

To display information about an IPv6 MLDv1 group for a specific multicast address:

```
device# show ipv6 mld groups ff1e::1
```

To display information about all IPv6 MLDv1 groups for a VLAN:

```
device# show ipv6 mld groups interface vlan 2000
```

To display summary information:

```
device# show ipv6 mld groups summary
MLD Route Summary of rbridge_id 1:
  No. of (*,G) Local Routes = 0
  No. of (S,G) Local Routes = 0
MLD Route Summary of rbridge_id 2:
  No. of (*,G) Local Routes = 0
  No. of (S,G) Local Routes = 0
```

## History

Release version	Command history
6.0.1	This command was modified to add the <b>summary</b> keyword.

# show ipv6 mld interface

Displays IPv6 MLDv1 snooping information for a VLAN interface.

## Syntax

```
show ipv6 mld interface { vlan vlan_id } [ rbridge-id { all | rbridge-id } ]
```

## Parameters

**vlan** *vlan\_id*

Specifies a VLAN ID. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

To display information about IPv6 MLDv1 snooping for a specific VLAN interface:

```
device# show ipv6 mld interface vlan 100
Interface Vlan 100
  MLD Snooping enabled
  MLD Snooping fast-leave disabled
  MLD Snooping restrict-unknown-multicast disabled
  MLD Snooping querier disabled
  MLD Snooping query interval is 125 Seconds
  MLD Snooping max query response time is 10 Seconds
  MLD Snooping Last member query response interval is 1 Seconds
  MLD Snooping last-member query count 2
  MLD Snooping startup-query interval 31 Seconds
  MLD Snooping startup-query count 2
  MLD Snooping robustness variable 2
  Number of router-ports: 0
```

## History

Release version	Command history
7.1.0	This command was modified to include additional MLD snooping information.

# show ipv6 mld snooping

Displays information about the actively enabled IPv6 MLDv1 snooping mechanism and related configurations such as the active querier, the number of group-learned mrouter present, and other querier details.

## Syntax

```
show ipv6 mld snooping [ interface vlan vlan_id ] [ mrouter [ interface vlan vlan_id ] ]
```

## Parameters

### interface vlan

Specifies a VLAN ID.

*vlan\_id*

A VLAN ID. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

### mrouter

Specifies all multicast router statistics.

## Modes

Privileged EXEC mode

## Examples

To display comprehensive information about the IPv6 MLDv1 snooping mechanism and related configurations:

```
switch# show ipv6 mld snooping
```

To display information about IPv6 MLDv1 snooping for a specific VLAN:

```
switch# show ipv6 mld snooping interface vlan 2000
```

To display information about all IPv6 MLDv1 multicast router snooping:

```
switch# show ipv6 mld snooping mrouter
```

To display information about IPv6 MLDv1 multicast router snooping for a specific VLAN:

```
switch# show ipv6 mld snooping mrouter interface vlan 2000
```

# show ipv6 mld statistics

Displays IPv6 MLDv1 statistics for a VLAN.

## Syntax

```
show ipv6 mld statistics interface vlan vlan_id [ rbridge-id { rbridge-id|all } ]
```

## Parameters

### interface vlan

Specifies a VLAN ID.

*vlan\_id*

A VLAN ID. Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

To display information about IPv6 MLDv1 statistics for a specific VLAN interface:

```
switch# show ipv6 mld statistics interface vlan 1
MLD packet statistics for Rbridge Id 3 in Vlan vlan1
MLD Message type      Edge-Received   Edge-Sent   Edge-Rx-Errors   ISL-Received
General Query         0               0           0                 0 <<
Group Specific Query  0               0           0                 0
V1 Membership Report  0               0           0                 0
V2 Membership Report  0               0           0                 0
Group Leave           0               0           0                 0

MLD Error Statistics:
Checksum Error        0
Size_or_Range_Error  0
```

## History

Release version	Command history
7.1.0	This command was modified to include MLD statistics.



# show ipv6 nd interface

Displays information about the IPv6 Neighbor Discovery configuration on an interface.

## Syntax

```
show ipv6 nd interface
```

```
show ipv6 nd interface rbridge-id { all | rbridge-id }
```

```
show ipv6 nd interface prefix [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
```

```
show ipv6 nd interface [ <N>gigabitethernet [ rbridge-id / ] slot / port [ prefix ] | port-channel number }
```

```
show ipv6 nd interface ve vlan_id [ prefix ] [ rbridge-id { all | rbridge-id } ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel number

Specifies a port-channel interface. The range is from 1 through 6144.

### prefix

Displays prefix information.

**vrf** *vrf-name*

Specifies a VRF instance.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

### ve *vlan\_id*

Specifies the VLAN ID of a virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Examples

The following example illustrates the output of the **show ipv6 nd interface** command for an Ethernet interface.

```
device# show ipv6 nd interface tengigabitethernet 7/0/46
ICMPv6 ND Interfaces for VRF default-vrf
IPv6 address: 2ffe::1
Router-Advertisement active timers:
  Last Router-Advertisement sent: 00:01:25
  Next Router-Advertisement sent in: 00:07:06
Router-Advertisement parameters:
  Periodic interval: 200 to 600 seconds
  Send 'Managed Address Configuration' flag: false
  Send 'Other Stateful Configuration' flag: false
  Send 'Current Hop Limit' field: 64
  Send 'MTU' option value: 1500
  Send 'Router Lifetime' field: 1800 secs
  Send 'Reachable Time' field: 0 ms
  Send 'Retrans Timer' field: 0 ms
  Suppress RA: false
  Suppress MTU in RA: false
  Suppress All RA: false
Neighbor-Solicitation parameters:
  NS retransmit interval: 1 secs
  DAD Attempts: 2
  DAD expiry: 1 secs
  Neighbor Cache Expiry: 14400 secs
```

The following example illustrates the output of the **show ipv6 nd interface** command for an Ethernet interface with the **prefix** keyword specified.

```
device# show ipv6 nd interface tengigabitethernet 7/0/46 prefix
ICMPv6 ND Interfaces for VRF default-vrf
List of IPv6 Prefix advertised on Te 7/0/46
  Prefix : 3001::/64
  Enabled : Yes
  Valid lifetime : 2592000
  Preferred lifetime : 604800
  On-link : Yes
  Off-link : No
  Autonomous : Yes
  Prefix : 2ffe::/64
  Enabled : Yes
  Valid lifetime : Infinite
  Preferred lifetime : Infinite
  On-link : Yes
  Off-link : No
  Autonomous : Yes
```

The following example illustrates the output of the **show ipv6 nd interface port-channel** command.

```
device# show ipv6 nd interface port-channel 20
ICMPv6 ND Interfaces for VRF default-vrf
IPv6 address: fe80::3
Router-Advertisement active timers:
  Last Router-Advertisement sent: 00:00:00
  Next Router-Advertisement sent in: 00:00:00
Router-Advertisement parameters:
  Periodic interval: 200 to 600 seconds
  Send 'Managed Address Configuration' flag: false
  Send 'Other Stateful Configuration' flag: false
  Send 'Current Hop Limit' field: 64
  Send 'MTU' option value: 1500
  Send 'Router Lifetime' field: 1800 secs
  Send 'Reachable Time' field: 0 ms
  Send 'Retrans Timer' field: 0 ms
  Suppress RA: false
  Suppress MTU in RA: false
  Suppress All RA: false
Neighbor-Solicitation parameters:
  NS retransmit interval: 1 secs
  DAD Attempts: 2
  DAD expiry: 1 secs
Neighbor Cache Expiry: 14400 secs
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.

# show ipv6 nd suppression-cache

Displays IPv6 neighbor discovery (ND)-suppression information.

## Syntax

```
show ipv6 nd suppression-cache [ summary | vlan vlan-id ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### summary

Specifies summary format.

### vlan *vlan-id*

Specifies a VLAN interface.

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the results of the basic form of this command.

```
device# show ipv6 nd suppression-cache
Flags: L - Local Learnt Adjacency
       R - Remote Learnt Adjacency
       RS - Remote Static Adjacency
-----
Vlan   IP           Mac           Interface    Age           Flags
-----
Ve 110  110::110     0011.9400.0066 Te 3/0/2     00:00:05     L
Ve 110  110::111     0011.9400.0067 Tu 61441     Never         R
Ve 110  110::112     0011.9400.0068 Tu 61441     Never         RS
Ve 110  110::113     0011.9400.0069 Tu 61441     Never         RS
```

(Output truncated for brevity)

## History

Release version	Command history
7.0.0	This command was introduced.

# show ipv6 nd suppression-statistics

Displays IPv6 neighbor discovery (ND)-suppression statistics.

## Syntax

```
show ipv6 nd suppression-statistics [ vlan vlan-id ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

**vlan** *vlan-id*

Specifies a VLAN interface.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the results of the basic form of this command.

```
device# show ipv6 nd suppression-statistics
Vlan      Forwarded   Suppressed  Remote-arp Proxy
-----
110       0           117         0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show ipv6 nd suppression-status

Displays the IPv6 neighbor discovery (ND)-suppression status.

## Syntax

```
show ipv6 nd suppression-status [ vlan vlan-id ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

**vlan** *vlan-id*

Specifies a VLAN interface.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the results of the basic form of this command.

```
device# show ipv6 nd suppression-status
Vlan      Configuration  Evpn-Register  Operation
-----
1         Disabled      No              Inactive
100      Disabled      No              Inactive
110      Enabled       Yes             Active
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show ipv6 neighbor

Displays information for a neighbor in IPv6 Neighbor Discovery.

## Syntax

```
show ipv6 neighbor
show ipv6 neighbor rbridge-id { all | rbridge-id }
show ipv6 neighbor { dynamic | static } [ summary ] [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
show ipv6 neighbor <N>gigabitethernet [ rbridge-id / ] slot / port [ vrf vrf-name ]
show ipv6 neighbor { ipv6_address | ve vlan_id | summary } [ vrf vrf-name ] [ rbridge-id { all | rbridge-id } ]
show ipv6 neighbor port-channel number [ vrf vrf-name ] [ force-delete | no-refresh ]
show ipv6 neighbor vrf vrf-name [ rbridge-id { all | rbridge-id } ]
show ipv6 neighbor slot slot [ ipv6_address | vrf { vrf-name rbridge-id { all | rbridge-id } | rbridge-id [ all | rbridge-id ] }
```

## Parameters

### dynamic

Displays dynamic information.

### static

Displays static information.

### summary

Displays summary information.

### vrf vrf-name

Specifies a VRF instance.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**ve** *vlan\_id*

Specifies the VLAN ID of a virtual Ethernet (VE) interface.

**port-channel** *number*

Specifies a port-channel interface. The range is from 1 through 6144.

**slot** *slot*

Specifies a line card.

## Modes

Privileged EXEC mode

## Usage Guidelines

If leaked subnet routes are present, that information displays in the output.

## Examples

The following example illustrates the output of the **show ipv6 neighbor** command without keywords.

```
device# show ipv6 neighbor
Address          Mac-address      Interface      MacResolved    Age           Type
-----
2001::10        0010.9400.0066  Te 3/0/2      yes            00:00:13     Dynamic
2001::11        0010.9400.0067  Te 3/0/2      yes            00:00:13     Dynamic
2001::12        0010.9400.0068  Te 3/0/2      yes            00:00:13     Dynamic
2001::13        0010.9400.0069  Te 3/0/2      yes            00:00:13     Dynamic
2001::14        0010.9400.006a  Te 3/0/2      yes            00:00:13     Dynamic
2001::15        0010.9400.006b  Te 3/0/2      yes            00:00:13     Dynamic
2001::16        0010.9400.006c  Te 3/0/2      yes            00:00:13     Dynamic
2001::17        0010.9400.006d  Te 3/0/2      yes            00:00:13     Dynamic
2001::18        0010.9400.006e  Te 3/0/2      yes            00:00:13     Dynamic
```

The following example illustrates the output of the **show ipv6 neighbor slot** command.

```
device# show ipv6 neighbor slot 0
Total Neighbors : 100
Address          Mac-address      Interface      MacResolved    Type
-----
2001::10        0010.9400.0066  Te 0/2         yes            Dynamic
2001::11        0010.9400.0067  Te 0/2         yes            Dynamic
2001::12        0010.9400.0068  Te 0/2         yes            Dynamic
2001::13        0010.9400.0069  Te 0/2         yes            Dynamic
```

The following example illustrates the output of the **show ipv6 neighbor port-channel** command.

```
device# show ipv6 neighbor port-channel 20
Address          Mac-address      Interface      MacResolved    Age           Type
-----
fe80::52eb:1aff:fe13:962  50eb.1a13.0962  Po 20         yes            02:51:54     Dynamic
```

## History

Release version	Command history
7.0.0	This command was modified to support port-channels.
7.0.1	This command was modified to display leaked subnet routes.



# show ipv6 ospf

Displays OSPFv3 information.

## Syntax

```
show ipv6 ospf [ vrf name [ rbridge-id { rbridge-id | all } ] ]
```

## Parameters

### vrf name

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

### rbridge-id

Specifies an RBridge or all RBridges.

#### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

When the RBridge ID is not specified, the output from the local node is displayed.

When the RBridge ID is specified, data from the corresponding specified RBridge is displayed.

When **all** is specified, data from all nodes in the cluster is displayed.

## Examples

This example shows sample output from the show ipv6 ospf command, including BFD configuration information.

```
device# show ipv6 ospf

OSPFv3 Process number 0 with Router ID 0x01010101(1.1.1.1)
Running 0 days 0 hours 12 minutes 18 seconds
Number of AS scoped LSAs is 0
Sum of AS scoped LSAs Checksum is 00000000
External LSA Limit is 250000
Database Overflow Interval is 10
Database Overflow State is NOT OVERFLOWED
Route calculation executed 6 times
Pending outgoing LSA count 0
Authentication key rollover interval 300 seconds
Number of areas in this router is 1
High Priority Message Queue Full count: 0
BFD is enabled, BFD HoldoverInterval: 1
Graceful restart helper is enabled, strict lsa checking is disabled
Nonstop Routing is enabled- sw0# show ipv6 ospf
```

## History

Release version	Command history
6.0.1	This command was modified to include BFD configuration status.

## Related Commands

[bfd](#), [bfd holdover-interval](#), [ipv6 ospf bfd](#), [show ipv6 ospf neighbor](#)

# show ipv6 ospf area

Displays the OSPFv3 area table in a specified format.

## Syntax

```
show ipv6 ospf area [A.B.C.D][ decimal ][ all-vrfs ][ rbridge-id rbridge-id ][ vrf vrfname ]
```

## Parameters

*A.B.C.D*

Area address in dotted decimal format.

*decimal*

Area address in decimal format. Valid values range from 0 to 2147483647.

**all-vrfs**

Specifies all VRFs.

**rbridge-id** *rbridge-id*

Displays the information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

**vrf** *vrf name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

## Modes

Privileged EXEC mode

## Examples

The following is sample output from the **show ipv6 ospf area** command when no arguments or keywords are used:

```
switch# show ipv6 ospf area
Area 0.0.0.1 :
Authentication: Not Configured
Active interface(s)attached to this area: Ve 2001 Ve 2002
Inactive interface(s)attached to this area: None
Number of Area scoped LSAs is 38
Sum of Area LSAs Checksum is 0015da81
Statistics of Area 0.0.0.1:
SPF algorithm executed 46 times
SPF last updated: 8518 sec ago
Current SPF node count: 7
Router: 4 Network: 3
Maximum of Hop count to nodes: 4
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf database

Displays lists of information about different OSPFv3 link-state advertisements (LSAs).

## Syntax

```
show ipv6 ospf database [ advrtr A.B.C.D ] [ all-vrfs ] [ as-external ] [ extensive ] [ inter-prefix ] [ inter-router ] [ intra-prefix ]
  [ link decimal ] [ link-id decimal ] [ network ] [ prefix ipv6-addr ] [ rbridge-id rbridge-id ] [ router ] [ summary ] [ type-7 ]
  [ vrf vrfname ]
```

```
show ipv6 ospf database scope { area { A.B.C.D | decimal } | as | link } [ all-vrfs ] [ rbridge-id rbridge-id ] [ vrf vrfname ]
```

## Parameters

**advrtr** *A.B.C.D*

Displays LSAs by Advertising Router Id in dotted decimal format.

**all-vrfs**

Specifies all VRFs in the cluster.

**as-external**

Displays information about external LSAs.

**extensive**

Displays detailed lists of LSA information.

**inter-prefix**

Displays information about inter area prefix LSAs.

**inter-router**

Displays information about inter area router LSAs.

**intra-prefix**

Displays information about intra area router LSAs.

**link** *decimal*

Displays information about the link LSAs.

**link-id** *decimal*

Link-state ID that differentiates LSAs. Valid values range from 1 to 4294967295.

**network**

Displays information about the network LSAs.

**prefix**

Displays information on the intra-area-prefix LSAs.

*ipv6-addr*

IPv6 address in dotted-decimal notation.

**rbridge-id** *rbridge-id*

Displays the information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

**router**

Displays information about the router LSAs.

**summary**

Displays LSA summary information.

**type-7**

Displays information about the not so stubby area (NSSA) external LSAs.

**vrf vrf name**

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

**scope**

Displays LSA information by LSA scope.

**area**

Displays LSAs by scope within a specified area.

**as**

Displays autonomous system (AS) LSAs by scope.

**link**

Displays link LSAs by scope.

## Modes

Privileged EXEC mode

## Examples

The following is sample output from the **show ipv6 ospf database as-external** command using the **link-id** keyword:

```
device# show ipv6 ospf database as-external link-id 5
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
N/A          Extn 33      4.4.4.4      80000002 1044 acb8 36  Yes
Bits: E--
Metric: 0
Prefix Options:
Referenced LSType: 0
Prefix: 2001:2:16::/64
```

The following is sample output from the **show ipv6 ospf database inter-prefix** command using the **link-id** keyword:

```
device# show ipv6 ospf database inter-prefix link-id 5
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
128          Inap 5      125.125.125.125 80000035 548 fe72 36  Yes
Metric: 1
Prefix Options:
Prefix: 2001:2000:8192::/64
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
128          Inap 5      122.122.122.122 80000035 565 a7e8 36  Yes
Metric: 1
Prefix Options:
Prefix: 2001:2001::/64
```

The following is sample output from the **show ipv6 ospf database inter-router** command using the **link-id** keyword:

```
device# show ipv6 ospf database inter-router link-id 5
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0.0.0.0      Iap 5          125.125.125.125 80000035 879 c910 32  Yes
  Options: V6E---R--
  Metric: 1
  Destination Router ID: 125.2.32.125
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0.0.0.1      Iap 5          122.122.122.122 80000035 1611 bb3c 32  Yes
  Options: -----
  Metric: 2
  Destination Router ID: 125.2.32.125
```

The following is sample output from the **show ipv6 ospf database intra-prefix** command using the **link-id** keyword:

```
device# show ipv6 ospf database intra-prefix link-id 5
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0            Iap 0          10.10.10.10 80000002 1572 63a0 44  Yes
  Number of Prefix: 1
  Referenced LS Type: Router
  Referenced LS ID: 0
  Referenced Advertising Router: 10.10.10.10
  Prefix Options: Metric: 1
  Prefix: 2050::/64
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0            Iap 46440      4.4.4.4     80000001 1649 d85c 44  Yes
  Number of Prefix: 1
  Referenced LS Type: Network
  Referenced LS ID: 1548
  Referenced Advertising Router: 4.4.4.4
  Prefix Options: Metric: 0
  Prefix: 3010::/64
```

The following is sample output from the **show ipv6 ospf database link** command using the **link-id** keyword:

```
device# show ipv6 ospf database link link-id 5
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0            Link 145      10.10.10.10 80000001 1639 0117 56  Yes
  Router Priority: 1
  Options: V6E---R--
  LinkLocal Address: fe80::205:33ff:fe77:2452
  Number of Prefix: 1
  Prefix Options:
  Prefix: 2050::/64
```

The following is sample output from the **show ipv6 ospf database network** command:

```
device# show ipv6 ospf database network
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID   Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0         Net  1548          4.4.4.4     80000001 1707 7b53 32  Yes
Options: V6E---R--
Attached Router: 4.4.4.4
Attached Router: 10.10.10.10
```

The following is sample output from the **show ipv6 ospf database router** command:

```
device# show ipv6 ospf database router
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID   Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0         Rtr  0            4.4.4.4     80000139 2   4b9a 24  Yes
Capability Bits: ---EB
Options: V6E---R--
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID   Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
10        Rtr  0            4.4.4.4     80000001 2   e236 24  Yes
Capability Bits: ---EB
Options: V6---NR--
```

The following is sample output from the **show ipv6 ospf database type-7** command:

```
device# show ipv6 ospf database type-7
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID   Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
3         Typ7 3          10.10.10.10 80000001 5   6c95 28  Yes
Bits: E--
Metric: 10
Prefix Options:
Referenced LSType: 0
Prefix: ::/0
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID   Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
3         Typ7 2          10.10.10.10 80000001 5   8107 36  Yes
Bits: E--
Metric: 10
Prefix Options:
Referenced LSType: 0
Prefix: 2111::/64
```

The following is sample output from the **show ipv6 ospf database link-id** command:

```
device# show ipv6 ospf database link-id 6514
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID   Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0         Link 6514        10.10.10.10 80000001 1696 6e44 56  Yes
Router Priority: 1
Options: V6E---R--
LinkLocal Address: fe80::205:33ff:fe77:23ee
Number of Prefix: 1
Prefix Options:
Prefix: 3010::/64
```

The following is sample output from the **show ipv6 ospf database extensive** command:

```
device# show ipv6 ospf database extensive
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age Cksum Len  Sync
0            Link 145      10.10.10.10  80000001 1559 0117 56  Yes
  Router Priority: 1
  Options: V6E---R--
  LinkLocal Address: fe80::205:33ff:fe77:2452
  Number of Prefix: 1
  Prefix Options:
  Prefix: 2050::/64
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age Cksum Len  Sync
0            Link 6514      10.10.10.10  80000001 1592 6e44 56  Yes
  Router Priority: 1
  Options: V6E---R--
  LinkLocal Address: fe80::205:33ff:fe77:23ee
  Number of Prefix: 1
  Prefix Options:
  Prefix: 3010::/64
```

The following is sample output from the **show ipv6 ospf database summary** command:

```
device# show ipv6 ospf database summary
AS scope:
  ASExternal      Active      MaxAge
  ASExternal      6           0
Area 0 scope:
  Router          Active      MaxAge
  Router          2           0
  Network         1           0
  InterPrefix    1           0
  InterRouter     0           0
  Type7           0           0
  IntraPrefix     2           0
  Other           0           0
  Total           6           0
Interface scope (over 2 interfaces):
  Link           Active      MaxAge
  Link           3           0
  Grace          0           0
  Other          0           0
  Total          3           0
Area 1 scope:
  Router          Active      MaxAge
  Router          1           0
  Network         0           0
  InterPrefix    2           0
  InterRouter     1           0
  Type7           0           0
  IntraPrefix     1           0
  Other           0           0
  Total           5           0
Interface scope (over 1 interfaces):
  Link           Active      MaxAge
  Link           1           0
  Grace          0           0
  Other          0           0
  Total          1           0
Total: 21 LSAs, 21 Active LSAs, 0 MaxAge LSAs
```



The following is sample output from the **show ipv6 ospf database scope** command using the **area** keyword:

```
device# show ipv6 ospf database scope area 0
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0            Rtr 0            10.10.10.10 80000005 1135 4bf6 40  Yes
  Capability Bits: ---EB
  Options: V6E---R--
  Type: Transit Metric: 1
  Interface ID: 6514      Neighbor Interface ID: 1548
  Neighbor Router ID: 4.4.4.4
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0            Rtr 0            4.4.4.4      800000f1 1295 9156 40  Yes
  Capability Bits: ---E-
  Options: V6E---R--
  Type: Transit Metric: 1
  Interface ID: 1548      Neighbor Interface ID: 1548
  Neighbor Router ID: 4.4.4.4
```

The following is sample output from the **show ipv6 ospf database prefix** command:

```
device# show ipv6 ospf database prefix L5001::10/128
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID      Adv Rtr      Seq(Hex) Age  Cksum Len  Sync
0            Iap 0            4.4.4.4      80000001 267 52a8 52  Yes
  Number of Prefix: 1
  Referenced LS Type: Router
  Referenced LS ID: 0
  Referenced Advertising Router: 4.4.4.4
  Prefix Options: LA, Metric: 0
  Prefix: 5001::10/128
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf interface

Displays interface information for all or specific OSPFv3-enabled interfaces.

## Syntax

**show ipv6 ospf interface**

**show ipv6 ospf interface brief** [ *all-vrfs* | *vrf vrf-name* ] [ *rbridge-id* { *rbridge-id* | *all* } ]

**show ipv6 ospf interface** [ *vrf vrf-name* ] [ *rbridge-id* { *rbridge-id* | *all* } ]

**show ipv6 ospf interface** <*N*>*gigabitethernet* [ *rbridge-id* / ] *slot* / *port*

**show ipv6 ospf interface** { *loopback number* | *port-channel number* | *ve vlan\_id* } [ *rbridge-id* { *rbridge-id* | *all* } ]

## Parameters

### **brief**

Displays brief summary about all enabled interfaces.

### **all-vrfs**

Displays the information for the physical, loopback, and SVI interfaces of all nodes in the VCS cluster. There can be multiple loopback or SVI interfaces with the same id from different rbridges. If *rbridge-id* is not specified the output will contain information from all vrfs from the local node only.

### **vrf** *vrf-name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

### **rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

### **all**

Specifies all RBridges.

### **<N>gigabitethernet**

Specifies a valid, physical Ethernet subtype ( <*N*> represents all available Ethernet speeds). Enter ? at the command prompt to see what interface subtypes are available for that command.

*rbridge-id*

Specifies the RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid slot number.

### **loopback** *number*

Specifies a loopback port number. The range is from 1 through 255.

### **port-channel** *number*

Specifies a port-channel interface. The range is from 1 through 6144.

**ve** *vlan\_id*

Specifies the VLAN number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **brief** keyword to limit the display to the following fields:

- Interface
- Area
- Status
- Type
- Cost
- State
- Nbrs(F/C)

## Examples

The following is sample output from the **show ipv6 ospf interface** command when no arguments or keywords are used.

```
device# show ipv6 ospf interface
vlan0.10 admin up, oper up, IPv6 enabled
IPv6 Address:fe80::205:33ff:fe77:23ee
3010::10/64
Instance ID 0, Router ID 10.10.10.10
Area ID 0, Cost 1, Type BROADCAST
MTU: 1500
State BDR, Transmit Delay 1 sec, Priority 1, Link-LSA Tx not suppressed
Timer intervals :
Hello 10, Hello Jitter 10 Dead 40, Retransmit 5
Authentication Use: Enabled
KeyRolloverTime(sec): Configured: 300 Current: 0
KeyRolloverState: NotActive
Outbound: None
Inbound: None
DR:4.4.4.4 BDR:10.10.10.10 Number of I/F scoped LSAs is 2
DRElection: 1 times, DelayedLSAck: 2 times
Neighbor Count = 1, Adjacent Neighbor Count= 1
Neighbor:
4.4.4.4 (DR)
Statistics of interface vlan0.10:
Type tx rx tx-byte rx-byte
Unknown 0 0 0 0
Hello 37 35 1476 1400
DbDesc 2 2 136 116
LSReq 1 1 52 64
LSUpdate 8 2 684 368
LSAck 3 8 208 348
OSPF messages dropped,no authentication: 0
```

show ipv6 ospf interface

The following is sample output from the **show ipv6 ospf interface** command the **brief** keyword is used.

```
device# show ipv6 ospf interface brief
Interface Area Status Type Cost State Nbrs (F/C)
te2/1      0    up   BCST 1   DR   0/0
vlan0.10  0    up   BCST 1   BDR  1/1
vlan0.20  1    up   BCST 1   DR   0/0
vlan0.30  1    up   P2P  10  P2P  0/0
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.0.0	This command was modified to support port-channels.

# show ipv6 ospf memory

Displays information about OSPFv3 memory usage.

## Syntax

```
show ipv6 ospf memory [ all-vrfs ] [ rbridge-id rbridge-id ] [ vrf vrfname ]
```

## Parameters

### **all-vrfs**

Displays the information for all VRF instances.

### **rbridge-id** *rbridge-id*

Displays the information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

### **vrf** *vrfname*

Displays the information for a specific VRF instance. If this option is not used, details for the default VRF are shown in the output.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display routes that have been redistributed into OSPFv3.

## Examples

The following is sample output from the **show ipv6 ospf memory** command when no arguments or keywords are used:

```
device# show ipv6 ospf memory
Total Dynamic Memory Allocated for this instance : 4551924 bytes
Memory Type           Size           Allocated   Max-alloc   Alloc-Fails
MTYPE_OSPF6_AREA     450720         2            4            0
MTYPE_OSPF6_AREA_RANGE 36              0            16           0
MTYPE_OSPF6_SUMMARY_ADDRE 32              0            16           0
MTYPE_OSPF6_IF       304            4            64           0
MTYPE_OSPF6_NEIGHBOR 12524          1            32           0
MTYPE_OSPF6_ROUTE_NODE 28             11           4096         0
MTYPE_OSPF6_ROUTE_INFO 44             11           4096         0
MTYPE_OSPF6_PREFIX   24             0            16           0
MTYPE_OSPF6_LSA      136            22           4096         0
MTYPE_OSPF6_VERTEX   176            4            64           0
MTYPE_OSPF6_SPTREE   48             2            2            0
MTYPE_OSPF6_NEXTHOP  32             7            256          0
MTYPE_OSPF6_EXTERNAL_INFO 44            0            4096         0
MTYPE_THREAD         36             13           1024         0
MTYPE_OSPF6_LINK_LIST 24             3224         20480        0
MTYPE_OSPF6_LINK_NODE 16             74           20480        0
MTYPE_OSPF6_LSA_RETRANSMI 12            0            8192         0
global memory pool for all instances
Memory Type           Size           Allocated   Max-alloc   Alloc-Fails
MTYPE_OSPF6_TOP       61480          1            1            0
MTYPE_OSPF6_LSA_HDR   56             22           23           0
MTYPE_OSPF6_RMAP_COMPILED 0              0            0            0
MTYPE_OSPF6_OTHER     0              0            0            0
MTYPE_THREAD_MASTER   84             1            1            0
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf neighbor

Displays detailed or summary OSPFv3 neighbor information.

## Syntax

```
show ipv6 ospf neighbor [ rbridge-id { rbridge-id | all } ]
show ipv6 ospf neighbor interface <N>gigabitethernet [ rbridge-id / ] slot / port
show ipv6 ospf neighbor interface { loopback number | port-channel number | ve vlan_id } [ rbridge-id { rbridge-id | all } ]
show ipv6 ospf neighbor [ all-vrfs | vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
show ipv6 ospf neighbor detail [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
show ipv6 ospf neighbor router-id A.B.C.D [ vrf vrf-name ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

all

Specifies all RBridges.

### interface

Displays OSPFv3 interface information.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

(Optional) Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### loopback *number*

Specifies a loopback port number. The range is from 1 through 255.

### port-channel *number*

Specifies a port-channel interface. The range is from 1 through 6144.

### ve *vlan\_id*

Specifies a virtual Ethernet (VE) interface.

### all-vrfs

Specifies information for all VRF instances.

**vrf** *vrf-name*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF instance are shown in the output.

**detail**

Specifies detailed neighbor information.

**router-id** *A.B.C.D*

Specifies neighbor information for the specified router ID (in dotted decimal format).

## Modes

Privileged EXEC mode

## Examples

The following example shows sample output from the **show ipv6 ospf neighbor** command when no arguments or keywords are used.

```
device# show ipv6 ospf neighbor
Total number of neighbors in all states: 1
Number of neighbors in state Full      : 1
RouterID      Pri State  DR          BDR          Interface    [State]
4.4.4.4       1 Full   4.4.4.4     10.10.10.10  vlan0.10    [BDR]
```

The following example shows sample output from the **show ipv6 ospf neighbor** command when the **detail** keyword is used.

```
device# show ipv6 ospf neighbor detail
Total number of neighbors in all states: 8
Number of neighbors in state Init      : 1
Number of neighbors in state Full      : 7
RouterID      Pri State  DR          BDR          Interface    [State]
153.153.153.153  1 Init   153.153.153.153 0.0.0.0     Fo 122/8/8   [DR]
Option: 00-00-00  QCount: 0   Timer: 37
125.125.125.125  1 Full   154.154.154.154 122.122.122.122 Ve 2000      [BDR]
Option: 00-00-13  QCount: 0   Timer: 2
154.154.154.154  1 Full   154.154.154.154 122.122.122.122 Ve 2000      [BDR]
Option: 00-00-13  QCount: 0   Timer: 38
122.21.21.122    0 Full   122.122.122.122 0.0.0.0     Ve 2001      [DR]
Option: 00-00-13  QCount: 0   Timer: 18
125.125.125.125  1 Full   122.122.122.122 125.125.125.125 Ve 2002      [DR]
Option: 00-00-13  QCount: 0   Timer: 5
125.125.125.125  1 Full   122.122.122.122 125.125.125.125 Ve 2009      [DR]
Option: 00-00-13  QCount: 0   Timer: 47
125.125.125.125  1 Full   128.128.128.128 125.125.125.125 Ve 2128      [DRot her]
Option: 00-00-13  QCount: 42  Timer: 420
128.128.128.128  1 Full   128.128.128.128 125.125.125.125 Ve 2128      [DRot her]
Option: 00-00-13  QCount: 0   Timer: 28
```

The following example shows sample output from the **show ipv6 ospf neighbor detail** command, including configured BFD information.

```
device# show ipv6 ospf neighbor detail
Total number of neighbors in all states: 1
Number of neighbors in state Full      : 1
RouterID      Pri State  DR          BDR          Interface    [State]
2.2.2.2       1 Full   2.2.2.2     1.1.1.1     Ve 11        [BDR]
Option: 00-00-13  QCount: 0   Timer: 686
BFD State: UP, BFD HoldoverInterval(sec):Configured: 3 Current: 0
```



## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	This command was modified to include BFD configuration status.
7.0.0	This command was modified to support port-channels.

## Related Commands

[bfd](#), [bfd holdover-interval](#), [ipv6 ospf bfd](#), [show ipv6 ospf](#)

# show ipv6 ospf redistribute route

Displays all IPv6 routes or a specified IPv6 route that the device has redistributed into OSPFv3.

## Syntax

```
show ipv6 ospf redistribute route [ A.B.C.D:M ] [ all-vrfs ] [ rbridge-id rbridge-id ] [ vrf vrfname ]
```

## Parameters

*A.B.C.D:M*

Specifies an IPv6 address.

**all-vrfs**

Displays all IPv6 routes that the device has redistributed into OSPFv3.

**rbridge-id** *rbridge-id*

Displays the information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

**vrf** *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

## Modes

Privileged EXEC mode

## Examples

The following is sample output from the **show ipv6 ospf redistribute route** command when no arguments or keywords are used:

```
device# show ipv6 ospf redistribute route
Id    Prefix          Protocol  Metric Type  Metric
2     2111::/64      Static   Type-2  10
1     3030::/64      Connect  Type-2   0
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf routes

Displays OSPFv3 routes.

## Syntax

```
show ipv6 ospf routes [A.B.C.D:M] [all-vrfs] [rbridge-id rbridge-id] [vrf vrfname]
```

## Parameters

*A.B.C.D:M*

Specifies a destination IPv6 address.

**all-vrfs**

Displays the entire OSPFv3 route table for a device.

**rbridge-id** *rbridge-id*

Displays the route information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

**vrf** *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

## Modes

Privileged EXEC mode

## Examples

To display OSPFv3-calculated routes for the default VRF:

```
device# show ipv6 ospf routes

Current Route count: 4
  Intra: 3 Inter: 0 External: 5 (Type1 0/Type2 5)
  Equal-cost multi-path: 0
  OSPF Type: IA- Intra, OA - Inter, E1 - External Type1, E2 - External Type2
Destination          Cost      E2Cost    Tag      Flags    Dis
E2 2014::/64         1         0         0        00000003 110
Next_Hop_Router      Outgoing_Interface Adv_Router
fe80::768e:f8ff:fe2a:4900
vlan0.10             4.4.4.4
Destination          Cost      E2Cost    Tag      Flags    Dis
E2 2015::/64         1         0         0        00000003 110
Next_Hop_Router      Outgoing_Interface Adv_Router
fe80::768e:f8ff:fe2a:4900
vlan0.10             4.4.4.4
Destination          Cost      E2Cost    Tag      Flags    Dis
IA 2050::/64         1         0         0        00000003 110
Next_Hop_Router      Outgoing_Interface Adv_Router
::                   te2/1          10.10.10.10
Destination          Cost      E2Cost    Tag      Flags    Dis
IA 3010::/64         1         0         0        00000003 110
Next_Hop_Router      Outgoing_Interface Adv_Router
::                   vlan0.10       4.4.4.4
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf spf

Displays OSPFv3 SPF node, table, and tree information.

## Syntax

```
show ipv6 ospf spf { node | table | tree } [ all-vrfs ] [ area { A.B.C.D | decimal } ] [ rbridge-id rbridge-id ] [ vrf vrfname ]
```

## Parameters

### node

Displays OSPFv3 node information.

### table

Specifies a SPF table.

### tree

Specifies a SPF tree.

### all-vrfs

Displays the information for all VRF instances.

### area

Specifies an area.

*A.B.C.D*

Area address in dotted decimal format.

*decimal*

Area address in decimal format.

### rbridge-id *rbridge-id*

Displays the information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

### vrf *vrfname*

Displays the information for a specific VRF instance. If this option is not used, details for the default VRF are shown in the output.

## Examples

The following is sample output from the **show ipv6 ospf spf** command when the **node** keyword is used:

```
device# show ipv6 ospf spf node
SPF node for Area 0
  SPF node 10.10.10.10, cost: 0, hops: 0
    nexthops to node:
      parent nodes:
        child nodes: 4.4.4.4:1548
  SPF node 4.4.4.4:1548, cost: 1, hops: 1
    nexthops to node:    :: vlan0.10
      parent nodes: 10.10.10.10
        child nodes: 4.4.4.4:0
  SPF node 4.4.4.4:0, cost: 1, hops: 2
    nexthops to node:    fe80::768e:f8ff:fe2a:4900 vlan0.10
      parent nodes: 4.4.4.4:1548
        child nodes:
SPF node for Area 1
  SPF node 10.10.10.10, cost: 0, hops: 0
    nexthops to node:
      parent nodes:
        child nodes:
```

The following is sample output from the **show ipv6 ospf spf** command when the **table** keyword is used:

```
device# show ipv6 ospf spf table
SPF table for Area 0
R 4.4.4.4          ---E- V6E---R--    1 fe80::768e:f8ff:fe2a:4900 vlan0.10
N 4.4.4.4[1548]  ----- V6E---R--    1 ::                               vlan0.10
  Destination      Bits Options Cost  Nexthop                               Interface
SPF table for Area 1
  Destination      Bits Options Cost  Nexthop                               Interface
```

The following is sample output from the **show ipv6 ospf spf** command when the **tree** keyword is used:

```
device# show ipv6 ospf spf tree
SPF tree for Area 0
+- 10.10.10.10 cost 0
  +- 4.4.4.4:1548 cost 1
    +- 4.4.4.4:0 cost 1

SPF tree for Area 1
+- 10.10.10.10 cost 0
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf summary

Displays summary information for all OSPFv3 instances.

## Syntax

```
show ipv6 ospf summary [ all-vrfs ] [ rbridge-id rbridge-id ] [ vrf vrfname ]
```

## Parameters

### all-vrfs

Specifies all VRF instances.

### rbridge-id *rbridge-id*

Displays the information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

### vrf *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

## Modes

Privileged EXEC mode

## Examples

The following is sample output from the **show ipv6 ospf summary** command when no arguments or keywords are used:

```
device# show ipv6 ospf summary

Total number of IPv6 OSPF instances: 1
Seq Instance      Intfs  Nbrs  Nbrs-Full LSAs  Routes
1  default-vrf    3      1     1      21     9
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf virtual-links

Displays information about all OSPFv3 virtual links or specified links.

## Syntax

```
show ipv6 ospf virtual-links [ all-vrfs ] [ brief ] [ rbridge-id rbridge-id ] [ vrf vrfname ]
```

## Parameters

### **all-vrfs**

Specifies all virtual links.

### **brief**

Displays brief summary information.

### **rbridge-id** *rbridge-id*

Displays information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

### **vrf** *vrfname*

Specifies a non-default VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display brief or detailed information about virtual links. You can show information about all virtual links, or you can specify a virtual link

Use the **brief** keyword to limit the display to the following fields:

- Index
- Transit Area ID
- Router ID
- Interface Address
- State



## Examples

The following is sample output from the **show ipv6 ospf virtual-links** command when no arguments or keywords are used:

```
device# show ipv6 ospf virtual-links
Transit Area ID Router ID      Interface Address      State
1               4.4.4.4                3010::10              P2P
  Timer intervals(sec) :
    Hello 10, Hello Jitter 10, Dead 40, Retransmit 5, TransmitDelay 1
  DelayedLSAck:      3 times
  Authentication: Not Configured
  Statistics:
    Type   tx      rx      tx-byte  rx-byte
    Unknown 0       0       0        0
    Hello   3       4       120      156
    DbDesc  2       3       156      124
    LSReq   1       1       40       76
    LSUpdate 7       3       656     240
    LSAck   3       4       148     224
  OSPF messages dropped,no authentication: 0
  Neighbor: State: Full Address: 3010::1 Interface: vlan0.10
```

The following is sample output from the **show ipv6 ospf virtual-links** command when the **brief** keyword is used:

```
device# show ipv6 ospf virtual-links brief
Transit Area ID Router ID      Interface Address      State
1               4.4.4.4                3010::10              P2P
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 ospf virtual-neighbor

Displays information about all OSPFv3 virtual neighbors or one OSPFv3 virtual neighbor that you specify.

## Syntax

```
show ipv6 ospf virtual-neighbor [ all-vrfs ] [ brief ] [ rbridge-id rbridge-id ] [ vrf vrfname ]
```

## Parameters

### all-vrfs

Specifies all virtual neighbors.

### brief

Displays brief summary information.

### rbridge-id *rbridge-id*

Displays information for the physical, loopback, and SVI interfaces specific to the selected RBridge.

### vrf *vrfname*

Specifies the name of the VRF instance. If this option is not used, details for the default VRF are shown in the output.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display brief or detailed information about virtual neighbors. You can show information about all virtual neighbors, or you can specify a virtual neighbor

Use the **brief** keyword to limit the display to the following fields:

- Index
- Router ID
- Address
- State
- Interface

## Examples

The following is sample output from the **show ipv6 ospf virtual-neighbor** command when no arguments or keywords are used:

```
device# show ipv6 ospf virtual-neighbor
Index Router ID      Address           State      Interface
 1     4.4.4.4          3010::1         Full      vlan0.10
Option: 00-00-00      QCount: 0      Timer: 457
```

The following is sample output from the **show ipv6 ospf virtual-neighbor** command when the brief keyword is used:

```
device# show ipv6 ospf virtual-neighbor brief
Index Router ID      Address          State   Interface
1      4.4.4.4           3010::1        Full   vlan0.10
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show ipv6 prefix-list

To displays the IPv6 prefix-list information, use the **show ipv6 prefix-list** command.

## Syntax

```
show ipv6 prefix-list [ name [ rbridge-id number ] | rbridge-id list_name ]
```

## Parameters

*name*

The name of the specific prefix-list you want to display.

**rbridge-id***number*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

Typical command output:

```
switch# show ipv6 prefix-list
ipv6 prefix-list routesfor2001: 1 entries
seq 5 permit 2001::/16
```

## Related Commands

[ipv6 prefix-list](#)

# show ipv6 raguard

Displays RA Guard status on a specified interface or on all device interfaces.

## Syntax

```
show ipv6 raguard
```

```
show ipv6 raguard [ interface { <N>gigabitethernet rbridge-id//slot/port | port-channel index } ]
```

## Parameters

### interface

Displays RA Guard information for an interface that you specify.

### <N> gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### port-channel *number*

Specifies a port-channel.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **show ipv6 raguard** form of this command displays the RA Guard status on all device interfaces.

The **show ipv6 raguard interface** form of this command displays the RA Guard status on a specific interface.

## Command Output

The **show ipv6 raguard** command displays the following information:

Output field	Description
Active	The configuration is active and all necessary hardware resources are configured.
Partial	The configuration is partially programmed and the configured action is taken in some cases. This is typically seen for logical interfaces like port channels, which span multiple hardware

Output field	Description
	resources. Some hardware resources may not have the necessary resources to support the request.
In progress	The configuration is currently being programmed into the hardware.
Inactive	The configuration is inactive and is not programmed in the hardware. This is typically seen when the hardware resources limit is reached.

## Examples

The following example displays the RA Guard status on all device interfaces.

```
device# show ipv6 raguard

RA -Guard is enabled on following interface (s):
  TenGigabitEthernet 2/2/1 (active)
  TenGigabitEthernet 2/2/2 (inactive)
  Port-channel 11 (partial)
```

The following example displays the RA Guard status on a specified port-channel.

```
device# show ipv6 raguard interface port-channel 11

Port-channel 11 (partial)
```

## History

Release version	Command history
6.0.0	This command was introduced.

## Related Commands

[ipv6 raguard](#)

# show ipv6 route

Displays information about IPv6 routes.

## Syntax

```
show ipv6 route [ ipv6address/prefix ] [ longer [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route [ ipv6address/prefix ] [ detail [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route [ ipv6address/prefix ] [ rbridge-id { all | rbridge-id } ] ]
show ipv6 route [ ipv6address/prefix ] [ vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route all [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route bgp [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route connected [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route detail [ rbridge-id { rbridge-id | all } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route import [ src-vrf-name ] [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route nexthop [ nexthop-id [ ref-routes ] ] [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route ospf [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route rbridge-id { all | rbridge-id }
show ipv6 route slot slot [ ipv6address | ipv6prefix | rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route static [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route summary [ rbridge-id { all | rbridge-id } | vrf vrf-name [ rbridge-id { rbridge-id | all } ] ] ]
show ipv6 route system-summary [ rbridge-id { all | rbridge-id } ]
show ipv6 route vrf vrf-name [ rbridge-id { all | rbridge-id } ] ]
```

## Parameters

*ipv6address/prefix*

IPv6 address and optional prefix in A:B::C:D /L format, where "L" is the prefix length.

**longer**

Displays routes that match the specified prefix.

**rbridge-id**

Specifies an RBridge or all RBridges (in context).

*rbridge-id*

Specifies an RBridge ID when followed by a valid number.

**all**

Specifies all RBridges (in context).

**vrf vrf-name**

Specifies a VRF instance.

*src-vrf-name*

Specifies the name of a source VRF instance.

*ipv6prefix*

IPv6 prefix in *A::B::C:D /L* format.

**all**

Specifies information about all routes.

**bgp**

Specifies information about BGP routes.

**connected**

Specifies information about directly connected routes.

**detail**

Specifies detailed information about routes.

**import**

Specifies imported IPv6 routes.

*src-vrf-name*

Specifies a VRF instance from which routes are leaked.

**nexthop** *nexthop-id*

Specifies a next-hop address.

*nexthop-id*

Next-hop ID. Range is from 0 through 4294967294.

**ref-routes**

Specifies routes that match the next-hop ID.

**ospf**

Specifies routes learned from the Open Shortest Path First (OSPF) protocol.

**slot** *line\_card\_number*

Specifies routes with the provided line card number.

**static**

Specifies static routes.

**summary**

Specifies summary information.

**system-summary**

Specifies summary system information.

**vrf** *vrf-name*

Specifies a VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

If leaked subnet routes are present, that information displays in the output.



## Examples

Typical command output for the **show ipv6 route** command:

```
device# show ipv6 route
IPv6 Routing Table for VRF "default-vrf"
Total number of IPv6 routes: 3
'[x/y]' denotes [preference/metric]

3:5::/64, attached
  via ::, Ve 3502, [0/0], 2d22h, direct, tag 0
3:5::2/128, attached
  via ::, Ve 3502, [0/0], 2d22h, local, tag 0
fe80::/64, attached
  via ::, , [0/0], 2d22h, local, tag 0
```

Typical command output for the **show ipv6 route detail** command:

```
device# show ipv6 route detail
IPv6 Routing Table for VRF "default-vrf"
Total number of IPv6 routes: 7
'*' denotes best ucast next-hop
'[x/y]' denotes [preference/metric]

33::/64, attached
  *via ::, Te 89/0/8, [0/0], 2m2s, direct, tag 0
33::33/128, attached
  *via ::, Te 89/0/8, [0/0], 2m2s, local, tag 0
44::/64, attached
  *via ::, Te 89/0/9, [0/0], 1m25s, direct, tag 0
44::44/128, attached
  *via ::, Te 89/0/9, [0/0], 1m25s, local, tag 0
55::/64, attached
  *via ::, Te 89/0/9, [33/1], 0m37s, static, tag 4294967295
fe80::/10, attached
  *via ::, , [0/0], 18h26m, local, tag 0
ff00::/8, attached
  *via ::, Null0, [0/0], 18h26m, local, tag 0
```

Typical command output for the **show ipv6 route summary vrf** command:

```
device# show ipv6 route summary vrf red
IPv6 Routing Table - 26 entries:
  8 direct, 0 static, 0 RIP, 0 OSPF, 8 BGP, 0 ISIS, 80 EVPN Host
Number of prefixes:
  /8: 1  /10: 1  /64: 16  /128: 88
Nexthop Table Entry - 2 entries
```

Typical command output for the **show ipv6 route system-summary** command:

```
device# show ipv6 route system-summary
System Route Count: 97 Max routes: 1024 (Route limit not exceeded)
System Nexthop Count: 2 Max nexthops: 1024 (Nexthop limit not exceeded)
One Path Nexthop Count: 3 Max One Path Nexthops: 5120

VRF-Name: blue
  Route count: 18 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 EVPN Host

VRF-Name: default-vrf
  Route count: 23 Max routes: Not Set (Route limit not exceeded)
  3 connected, 0 static, 0 RIP, 0 OSPF, 16 BGP, 0 ISIS, 0 EVPN Host

VRF-Name: green
  Route count: 18 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 EVPN Host

VRF-Name: mgmt-vrf
  Route count: 2 Max routes: Not Set (Route limit not exceeded)
  0 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 EVPN Host

VRF-Name: red
  Route count: 18 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 EVPN Host

VRF-Name: yellow
  Route count: 18 Max routes: Not Set (Route limit not exceeded)
  8 connected, 0 static, 0 RIP, 0 OSPF, 0 BGP, 0 ISIS, 0 EVPN Host
```

## History

Release version	Command history
6.0.0	This command was modified to support the <b>vrf</b> keyword.
6.0.1a	This command was modified to support the <b>detail</b> keyword.
7.0.1	This command was modified to display leaked subnet routes and total number of routes. One Path Nexthop Count option added.
7.2.0	The command output display was enhanced.

# show ipv6 static route

Displays information about IPv6 static routes.

## Syntax

```
show ipv6 static route [ ipv6prefix [ rbridge-id { all | rbridge-id } ] [ vrf vrf-name ] | rbridge-id { all | rbridge-id } | vrf vrf-name ]
```

## Parameters

*ipv6prefix*

IPv6 prefix in *A:B::/length* format.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges in the cluster.

**vrf** *vrf-name*

Specifies a VRF instance.

**all**

Specifies all VRF instances.

**default**

Specifies the default VRF instance.

## Modes

Privileged EXEC mode

## Examples

The following example illustrates the output of the **show ipv6 static route** command:

```
switch# show ipv6 static route
IPv6 Configured Static Routes for VRF "default-vrf"
2014::/64-> 3001:1111::2000 preference: 1
  nh_vrf (default-vrf)
3ffe:1002::/64-> 2001::100 preference: 1
  nh_vrf (default-vrf)
  real-next-hop: 2001::100, interface: Te 7/0/48
2ffe:1111:2222::1234/128-> 3001::5678 preference: 1
  nh_vrf (default-vrf)
```

# show ipv6 vrrp

Displays information about IPv6 VRRP and VRRP-E sessions.

## Syntax

```
show ipv6 vrrp [ rbridge-id { rbridge-id | all } ]
show ipv6 vrrp VRID [ detail | summary ] [ rbridge-id { rbridge-id | all } ]
show ipv6 vrrp detail [ rbridge-id { rbridge-id | all } ]
show ipv6 vrrp summary [ vrf { vrf-name | all | default-vrf } | rbridge-id { rbridge-id | all } ]
show ipv6 vrrp interface [ <N>gigabitethernet [ rbridge-id / slot / port | port-channel number ] [ detail | summary ]
show ipv6 vrrp interface ve vlan_id [ detail | summary ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### *VRID*

The virtual group ID about which to display information. The range is from 1 through 255.

### detail

Displays all session information in detail, including session statistics.

### summary

Displays session-information summaries.

### vrf

Specifies a VRF instance or all VRFs.

#### *vrf-name*

Specifies a VRF instance. For the default vrf, enter **default-vrf**.

### all

Specifies all VRFs.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

### interface

Displays information for an interface that you specify.

### <N> gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

(Optional) Specifies an RBridge ID.

<i>slot</i>	Specifies a valid slot number.
<i>port</i>	Specifies a valid port number.
<b>ve</b> <i>vlan_id</i>	Specifies the VE VLAN number.
<b>port-channel</b> <i>number</i>	Specifies a port-channel interface. The range is from 1 through 6144.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display information about IPv6 VRRP and VRRP-E sessions, either in summary or full-detail format. You can also specify a particular virtual group, one or all RBridge IDs, or an interface for which to display VRRP output.

### NOTE

IPv6 VRRP-E supports only the VE interface type.

To display information for IPv6 VRRP sessions using the default VRF, you can use the **show ipv6 vrrp summary** syntax (with no additional parameters).

To display information for the default or a named VRF, you can use the **show ipv6 vrrp summary vrf** syntax with the *vrf-name* option.

To display information about all VRFs, use the **show ipv6 vrrp summary vrf all** syntax.

## Examples

The following example displays information about all IPv6 VRRP/VRRP-E sessions on the device.

```
device# show ipv6 vrrp

=====
Rbridge-id:4
=====

Total number of VRRP session(s)   : 2

VRID 18
  Interface: Ve 2018;  Ifindex: 1207961570
  Mode: VRRP
  Admin Status: Enabled
  Description :
  Address family: IPv6
  Version: 3
  Authentication type: No Authentication
  State: Master
  Session Master IP Address: Local
  Virtual IP(s): fe80::1
  Configured Priority: unset (default: 100); Current Priority: 100
  Advertisement interval: 1000 milli sec (default: 1000 milli sec)
  Preempt mode: ENABLE (default: ENABLE)
  Hold time: 0 sec (default: 0 sec)
  Trackport:
    Port(s)                Priority  Port Status
    =====                =====  =====
  Statistics:
    Advertisements: Rx: 0, Tx: 35
    Neighbor Advertisements: Tx: 1

VRID 19
  Interface: Ve 2019;  Ifindex: 1207961571
  Mode: VRRP
  Admin Status: Enabled
  Description :
  Address family: IPv6
  Version: 3
  Authentication type: No Authentication
  State: Master
  Session Master IP Address: Local
  Virtual IP(s): fe80::1
  Configured Priority: unset (default: 100); Current Priority: 100
  Advertisement interval: 1000 milli sec (default: 1000 milli sec)
  Preempt mode: ENABLE (default: ENABLE)
  Hold time: 0 sec (default: 0 sec)
  Trackport:
    Port(s)                Priority  Port Status
    =====                =====  =====
  Statistics:
    Advertisements: Rx: 0, Tx: 448
    Neighbor Advertisements: Tx: 1
```

The following example displays IPv6 VRRP/VRRP-E information in detail for a specific virtual group ID of 19, including session statistics.

```
device# show ipv6 vrrp 19 detail
=====Rbridge-id:122=====
Total number of VRRP session(s)   : 1
VRID 19
  Interface: Ve 2019; Ifindex: 1207961571
  Mode: VRRPE
  Admin Status: Enabled
  Description :
  Address family: IPv6
  Version: 3
  Authentication type: No Authentication
  State: Backup
  Session Master IP Address: fe80::205:33ff:fe79:fb1e
  Virtual IP(s): 2001:2019:8192::1
  Virtual MAC Address: 02e0.5200.2513
  Configured Priority: unset (default: 100); Current Priority: 100
  Advertisement interval: 1 sec (default: 1 sec)
  Preempt mode: DISABLE (default: DISABLED)
  Advertise-backup: ENABLE (default: DISABLED)
  Backup Advertisement interval: 60 sec (default: 60 sec)
  Short-path-forwarding: Enabled
  Revert-Priority: unset; SPF Reverted: No
  Hold time: 0 sec (default: 0 sec)
  Master Down interval: 4 sec
  Trackport:
    Port(s)                Priority  Port Status
    =====
Global Statistics:
=====
Checksum Error : 0
Version Error  : 0
VRID Invalid   : 0
Session Statistics:
=====
Advertisements      : Rx: 103259, Tx: 1721
Neighbor Advertisements : Tx: 0
Session becoming master : 0
Advts with wrong interval : 0
Prio Zero pkts      : Rx: 0, Tx: 0
Invalid Pkts Rvcd   : 0
Bad Virtual-IP Pkts : 0
Invalid Authenticon type : 0
Invalid TTL Value   : 0
Invalid Packet Length : 0
VRRPE backup advt sent : 1721
VRRPE backup advt recvd : 0
```

The following example displays summary information for IPv6 VRRP/VRRP-E statistics on the default VRF. (This command is equivalent to **show ipv6 vrrp summary vrf default-vrf**.)

```
device# show ipv6 vrrp summary
=====
Rbridge-id:4
=====
Total number of VRRP session(s)   : 1
Master session count   : 1
Backup session count   : 0
Init session count     : 0

VRID  Session  Interface  Admin  Current  State  Short-path  Revert  SPF
=====  =====  =====  =====  =====  =====  =====  =====  =====
19     VRRPE     Ve 2019   Enabled 100      Master  Enabled    unset   No
```

The following example displays summary information for IPv6 VRRP/VRRP-E statistics on the VRF named red.

```
device# show ipv6 vrrp summary vrf red
=====
Rbridge-id:4
=====

Total number of VRRP session(s) : 1
Master session count : 1
Backup session count : 0
Init session count : 0

VRID  Session  Interface  Admin  Current  State  Short-path  Revert  SPF
====  =====  =====  =====  =====  =====  =====  =====  =====
18    VRRPE     Ve 2018   Enabled 100     Master  Enabled     unset   No
```

The following example displays summary information for IPv6 VRRP/VRRP-E statistics on all VRFs.

```
device# show ipv6 vrrp summary vrf all
=====
Rbridge-id:4
=====

Total number of VRRP session(s) : 2
Master session count : 2
Backup session count : 0
Init session count : 0

VRID  Session  Interface  Admin  Current  State  Short-path  Revert  SPF
====  =====  =====  =====  =====  =====  =====  =====  =====
18    VRRPE     Ve 2018   Enabled 100     Master  Enabled     unset   No
19    VRRPE     Ve 2019   Enabled 100     Master  Enabled     unset   No
```



The following example displays information for IPv6 VRRP-E tracked networks.

```

device# show ipv6 vrrp detail

=====
Rbridge-id:1
=====

Total number of VRRP session(s)   : 1

VRID 2
  Interface: Ve 100;  Ifindex: 1207959652
  Mode: VRRPE
  Admin Status: Enabled
  Description :
  Address family: IPv6
  Version: 3
  Authentication type: No Authentication
  State: Master
  Session Master IP Address: Local
  Virtual IP(s): 2001:2019:8192::1
  Virtual MAC Address: 02e0.5225.1002
  Configured Priority: unset (default: 100); Current Priority: 100
  Advertisement interval: 1 sec (default: 1 sec)
  Preempt mode: DISABLE (default: DISABLED)
  Advertise-backup: DISABLE (default: DISABLED)
  Backup Advertisement interval: 60 sec (default: 60 sec)
  Short-path-forwarding: Disabled
  Revert-Priority: unset; SPF Reverted: No
  Hold time: 0 sec (default: 0 sec)
  Master Down interval: 4 sec
  Trackport:
    Port(s)                Priority  Port Status
    =====                =====  =====
  Tracknetwork:
    Network(s)              Priority  Status
    =====                =====  =====
    2001::/64                20       Up

Global Statistics:
=====
  Checksum Error : 0
  Version Error  : 0
  VRID Invalid   : 0

Session Statistics:
=====
  Advertisements           : Rx: 0, Tx: 132
  Neighbor Advertisements  : Tx: 66
  Session becoming master  : 1
  Advts with wrong interval : 0
  Prio Zero pkts           : Rx: 0, Tx: 0
  Invalid Pkts Rcvd        : 0
  Bad Virtual-IP Pkts      : 0
  Invalid Authentication type : 0
  Invalid TTL Value        : 0
  Invalid Packet Length    : 0
  VRRPE backup advt sent   : 0
  VRRPE backup advt recvd  : 0

```

## History

Release version	Command history
6.0.1	This command was modified to add output to verify the VRRP-E track network feature.
7.0.0	This command was modified to support port-channels.

show ipv6 vrrp

## Related Commands

[clear ipv6 vrrp statistics](#), [show vrrp](#)

# show lacp

Displays Link Aggregation Control Protocol (LACP) statistics.

## Syntax

```
show lacp [ counters [ port-channel ] | sys-id [ port-channel ]
```

## Parameters

### counters

Displays LACP statistics for all port-channel interfaces.

### *port-channel*

Displays counters for a specified port channel interface. Valid values range from 1 through 6144.

### sys-id

Displays LACP statistics by system ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the LACP statistics for each port-channel interface for all port-channel interfaces or a single port-channel interface, or by system ID.

# show lacp sys-id

Displays the Link Aggregation Control Protocol (LACP) system ID and priority information.

## Syntax

```
show lacp sys-id
```

## Modes

Privileged EXEC mode

## Usage Guidelines

The system priority and the system Media Access Control (MAC) address make up the system identification. The first two bytes are the system priority, and the last six bytes are the globally administered individual MAC addresses associated with the system.

## Examples

To display the local system ID:

```
switch# show lacp sys-id
% System 8000,00-05-1e-76-1a-a6
```

# show license

Displays license information.

## Syntax

```
show license [ rbridge-id { rbridge-id | all } ] [ all ]
```

## Command Default

Displays the licenses installed on the local switch.

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

### all

Executes the command on all devices in the fabric.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the license information for any switch in a cluster. The command output includes the RBridge ID, license description, expiration if applicable, the feature name, and an indication of whether the license is valid. A string of "x" characters is displayed for the license key.

Remote license operations may be performed on any remote RBridge, from any RBridge in the cluster.

## Examples

The following example displays an Extreme VDX 8770 licensed for Advanced Services: (This configuration enables the use of Layer 3 and FCoE features. The VCS Fabric license is enabled on all VDX platforms by default starting with Network OS 4.1.0; a VCS Fabric license does not need to be installed to enable VCS Fabric functionality.)

```
device# show license

rbridge-id: 60
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Advanced Services license
Feature name:ADVANCED_SERVICES
License is valid
```

show license

## Related Commands

license add, license remove, show license id

# show license id

Displays the RBridge License ID.

## Syntax

```
show license id [ rbridge-id { rbridge-id { rbridge-id | all } } ] [ all ]
```

## Command Default

Displays the license ID installed on the local switch.

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

### all

Executes the command on all switches in the fabric.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the RBridge license ID (WWN) for the specified switch. You need the RBridge license ID when you prepare to add a license.

Remote license operations may be performed on any remote RBridge, from any RBridge in the cluster.

## Examples

To display the license ID for the local switch:

```
device# show license id

Rbridge-Id      LicenseId
=====
2                10:00:00:05:1E:00:4C:80
```

## Related Commands

[license add](#), [license remove](#), [show license](#)

# show linecard

Displays information about the line cards present in the chassis.

## Syntax

```
show linecard [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID. The range of valid values is from 1 through 239.

## Modes

Privileged EXEC mode

## Command Output

The **show linecard** command displays the following information:

Output field	Description
Slot	Displays the slot number. Slots for line cards are L1 through L4 on Extreme VDX 8770-4 switches, and L1 through L8 on Extreme VDX 8770-8 switches.
Type	Displays the line card type.
Description	Module description.
ID	Displays IDs for line cards.
Status	Displays the status of the line card as one of the following: <ul style="list-style-type: none"> <li>VACANT - The slot is empty.</li> <li>POWERED-OFF - The module is present in the slot but is powered off.</li> <li>POWERING UP - The module is present and powering on.</li> <li>LOADING - The module is present, powered on, and loading the initial configuration.</li> <li>DIAG RUNNING POST1 - The module is present, powered on, and running the POST (power-on self-test).</li> <li>DIAG RUNNING POST2 - The module is present, powered on, and running the reboot power on self tests.</li> <li>INITIALIZING - The module is present, powered on, and initializing hardware components.</li> <li>ENABLED - The module is on and fully enabled.</li> <li>DISABLED - The module is powered on but disabled.</li> <li>FAULTY - The module is faulty because an error was detected.</li> <li>UNKNOWN - The module is inserted but its state cannot be determined.</li> </ul>



## Examples

The following example displays the line cards in an Extreme VDX 8770-4 switch.

```
device# show linecard
```

Slot	Type	Description	ID	Status
L1	LC48X10G	48-port 10GE card	114	ENABLED
L2	LC48X10G	48-port 10GE card	114	ENABLED
L3				VACANT
L4	LC48X1G	48-port 1GE card	131	ENABLED

The following example displays the line cards in RBridge 80.

```
device# show linecard rbridge-id 80
Rbridge-id 80:
```

Slot	Type	Description	ID	Status
L1	LC6X100G	6-port 100GE card	149	ENABLED
L2	LC48X10G	48-port 10GE card	114	ENABLED
L3	LC48X10G	48-port 10GE card	114	ENABLED
L4	LC12X40G	12-port 40GE card	127	ENABLED

## History

Release version	Command history
6.0.0	This command was modified to support the <b>rbridge-id</b> parameter.

## Related Commands

[linecard](#), [show sfm](#), [show slots](#)

# show lldp

Displays the LLDP status and transmitted Type, Length, and Values (TLV) information.

## Syntax

```
show lldp
```

## Modes

Privileged EXEC mode

## Examples

The following example displays all LLDP information including BGP automatic neighbor discovery configurations.

```
device# show lldp

LLDP Global Information
system-name: SLX
system-description: DEFAULT Extreme VDX switch
description:
State: Enabled
Mode: Receive/Transmit
Advertise transmitted: 30 seconds
Hold time for advertise: 120 seconds
Tx Delay Timer: 1 seconds
Transmit TLVs: Chassis ID          Port ID
                TTL                Port Description
                System Name         IEEE DCBx
                BGP Auto NBR
DCBx iSCSI Priority Values: none
```

## History

Release version	Command history
7.2.0	This command was modified to include BGP automatic neighbor discovery information.

# show lldp interface

Displays the LLDP status on the specified interface.

## Syntax

```
show lldp interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

To display all the LLDP interface information for a selected interface:

```
device# show lldp interface tengigabitethernet 1/0/0

LLDP information for Te 1/0/0
State:                               Enabled
Mode:                                 Receive/Transmit
Advertise Transmitted:                30 seconds
Hold time for advertise:              120 seconds
Tx Delay Timer:                       1 seconds
DCBX Version :                        CEE
Auto-Sense :                           Yes
Transmit TLVs:                        Chassis ID          Port ID
                                       TTL                  System Name
                                       IEEE DCBx          DCBx FCoE App
                                       DCBx FCoE Logical Link
DCBX FCoE Priority Values: 3
DCBX iSCSI Priority Values: 4
```

To display all the LLDP interface information:

```
device# show lldp
LLDP Global Information
system-name: switch
system-description: VDX-VCS 100
description:
State:                               Enabled
Mode:                                 Receive/Transmit
Advertise transmitted:                30 seconds
Hold time for advertise:              120 seconds
Tx Delay Timer:                       1 seconds
Transmit TLVs:                        Chassis ID          Port ID
                                       TTL                  System Name
                                       IEEE DCBx          DCBx FCoE App
                                       DCBx FCoE Logical Link
DCBX FCoE Priority Values: 3
DCBX iSCSI Priority Values: 4
```

# show lldp neighbors

Displays LLDP information for all neighboring devices on the specified interface.

## Syntax

```
show lldp neighbors [ interface { <N>gigabitethernet rbridge-id/slot/port } detail ]
```

## Parameters

### interface

Specifies an Ethernet interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### detail

Displays all the LLDP neighbor information in detail for the specified interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display LLDP information for all neighboring devices on the specified interface.

### ATTENTION

If you do not use the **interface** parameter, only the mandatory TLVs are displayed.

## Examples

The following example displays detailed LLDP neighbor information on a specific interface.

```
device# show lldp neighbors interface tengigabitethernet 3/0/8 detail

Neighbors for Interface Te 3/0/8
MANDATORY TLVs
=====
Local Interface: Te 0/8      Remote Interface: Te 3/0/8 (IF Name)
Dead Interval: 120 secs  Remaining Life : 100 secs Tx: 536  Rx: 535
Chassis ID: 0005.1e76.1020 (MAC)
Remote Mac: 0005.1e76.102c
OPTIONAL TLVs
=====
Port Interface Description: Te 3/0/8
System Name: sw0
System Description: Fibre Channel Switch.
System Capabilities: Switching Routing
System Capabilities Enabled: Switching
Link Prim: 257
Remote Protocols Advertised: Multiple Spanning Tree Protocol
Remote VLANs Configured: VLAN ID: 1  VLAN Name: default
AutoNego Support: Supported Not Enabled
AutoNego Capability: 0
Operational MAU Type: 0
Link Aggregation Capability: Capable
Link Aggregation Status: Disabled
Port Vlan Id: 1
Port & Protocol Vlan Flag: Supported Not enabled
Port & Protocol Vlan Id: 0
Link Aggregation Port Id: 0
Max Frame Size: 2500
Management Address: 10.32.152.21 (IPv4)
Interface Numbering: 2
Interface Number: 0x4080100 (67633408)
OID: 0x100f99b4
```

The following example shows sample output from the **show lldp neighbors** command for a specified Ethernet interface, including BGP automatic neighbor discovery configurations, when the **detail** keyword is used.

```
device# show lldp neighbors interface ethernet 0/24 detail

Neighbors for Interface Eth 0/24

MANDATORY TLVs
=====
Local Interface: Eth 0/24  (Local Interface MAC: 609c.9f70.771c)
Remote Interface: Ethernet 0/53 (Remote Interface MAC: 609c.9f70.3539)
Dead Interval: 120 secs
Remaining Life : 99 secs
Chassis ID: 609c.9f70.3500
LLDP PDU Transmitted: 135  Received: 132

OPTIONAL TLVs
=====
Port Interface Description: Eth 0/53
System Name: SLX

DCBX TLVs
=====

BGP Auto NBR TLV
=====
Remote AS number: 2250
Remote peer address: 5.5.5.2
```

## History

Release version	Command history
7.2.0	This command was modified to include BGP automatic neighbor discovery information.

# show lldp statistics

Displays the LLDP statistics on all interfaces or a specified interface.

## Syntax

```
show lldp statistics [ interface { <N>gigabitethernet rbridge-id/slot/port } ]
```

## Parameters

### interface

Specifies an Ethernet interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

If you do not specify an interface, this command displays the LLDP statistics for all interfaces.

## Examples

To display LLDP statistics on the specified interface:

```
switch# show lldp statistics interface tengigabitethernet 5/0/8
```

```
LLDP Interface statistics for Te 5/0/8
Frames transmitted: 555
Frames Aged out:    0
Frames Discarded:  0
Frames with Error: 0
Frames Recieved:   554
TLVs discarded:    0
TLVs unrecognized: 0
```



# show logging auditlog

Displays the internal audit log buffer of the switch.

## Syntax

```
show logging auditlog [ count count ] [ reverse ] [ rbridge-id rbridge-id | all ]
```

## Parameters

**count** *count*

Specifies the number of messages to display.

**reverse**

Displays the audit log in reverse order.

**rbridge-id** *rbridge-id*

Specifies an RBridge.

**all**

Executes the command on all switches in the fabric.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

## Examples

To display the audit log messages stored in the internal buffer:

```
switch# show logging auditlog
```

```
0 AUDIT,2012/04/13-02:35:59 (GMT), [DCM-2002], INFO, DCMCFG, admin/admin/10.72.16.41/ssh/cli,, chassis,
Event: noscli exit, Status: success, Info: Successful logout by user [admin].
1 AUDIT,2012/04/13-02:43:23 (GMT), [DCM-2001], INFO, DCMCFG, admin/admin/10.72.16.41/ssh/cli,, chassis,
Event: noscli start, Status: success, Info: Successful login attempt through ssh from 10.72.16.41.
```

## Related Commands

[clear logging auditlog](#), [clear logging raslog](#), [log-dampening-debug](#)

# show logging raslog

Displays the internal RASLog buffer of the switch.

## Syntax

```
show logging raslog [ attribute attribute ] [ blade blade ] [ count count ] [ message-type type ] [ reverse ] [ severity severity ]  
[ rbridge-id rbridge-id ]
```

## Parameters

**attribute** *attribute*

Filters output by message attribute. Valid attributes include FFDC and VCS.

**blade** *blade*

Displays for the specified blade only. Valid values for blade include MM1, MM2, and LC[1-8].

**count** *count*

Specifies the number of messages to display.

**message-type** *type*

Filters the output by message type. Valid message types include DCE or SYSTEM.

**severity** *severity*

Filters the output by message severity. Valid severity levels include the following: critical, error, info, and warning.

**reverse**

Displays the messages in reverse order.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the filters to customize the output.

This command is supported only on the local switch.

The RASLog messages contain the module name, error code, and message details.

## Examples

To display all RASLog messages stored in the system:

```
switch# show logging raslog

NOS: 3.0.0
2012/05/25-17:37:15, [LOG-1003], 1, M1, INFO, VDX8770-4, SYSTEM error log has been cleared
2012/05/25-17:38:32, [SEC-1203], 3, M1, INFO, sw0, Login information: Login successful via TELNET/SSH/
RSH. IP Addr: 10.24.65.24
2012/05/25-17:42:54, [SEC-1203], 4, M1, INFO, sw0, Login information: Login successful via TELNET/SSH/
RSH. IP Addr: 10.24.65.24
2012/05/25-17:43:12, [IPAD-1002], 5, M1, INFO, VDX8770-4, Switch name has been successfully changed to
dutA1-sw0.
2012/05/25-17:51:42, [FW-1439], 180, M1, WARNING, dutA1-sw0, Switch status change contributing factor
Switch offline.
(Output truncated)
```

To display all RASLog messages for a line card:

```
switch# show logging raslog blade LC2

NOS: 3.0.0
2012/05/28-12:07:41, [HASM-1004], 822, L2, INFO, VDX8770-4, Processor rebooted - Reset
2012/05/28-12:07:41, [HASM-1104], 823, L2, INFO, VDX8770-4, Heartbeat to M1 up
2012/05/28-12:07:48, [HASM-1108], 830, L2, INFO, VDX8770-4, All service instances become active.
2012/05/29-13:32:50, [HASM-1004], 2721, L2, INFO, VDX8770-4, Processor rebooted - Reset
```

To display warning messages only on the standby management module:

```
switch# show logging raslog blade MM1 severity warning

NOS: 3.0.0
2012/03/09-15:20:55, [FW-1042], 26, M1, WARNING, dutA1-sw0, Sfp TX power for port 1/2/9, is below low
boundary(High=1999, Low=125). Current value is 17 uW.
2012/03/09-15:20:55, [FW-1046], 27, M1, WARNING, dutA1-sw0, Sfp Current for port 1/2/9, is below low
boundary(High=10, Low=3). Current value is 0 mA.
2012/03/09-15:20:55, [FW-1042], 28, M1, WARNING, dutA1-sw0, Sfp TX power for port 1/2/17, is below low
boundary(High=1999, Low=125). Current value is 18 uW.
(Output truncated)
```

To display only the FFDC messages:

```
switch# show logging raslog attribute FFDC rbridge-id 1

NOS: 3.0.0
1970/01/01-00:09:43, [HASM-1200], 106, MM1 | FFDC, WARNING, chassis, Detected termination of process
Dcmd.Linux.powe:1660
```

## Related Commands

[clear logging raslog](#), [logging raslog console](#)

## show mac-address-table

Displays forwarding information for all MAC addresses, for a specific dynamic or static MAC address, for all dynamic MAC addresses, for all static MAC addresses, for a specific interface, for a specific VLAN, for MAC addresses associated with port profiles, and for all MAC addresses in the Layer 2 forwarding database for BGP EVPN.

### Syntax

```
show mac-address-table [ address mac-addr | aging-time [ conversational [ rbridge-id ] rbridge-id ] ] | authenticated
interface <N> gigabitethernet rbridge-id/slot/port | authentication-failed interface <N> gigabitethernet rbridge-id/slot/
port | count [ address MAC_address | conversational linecard linecard_number [ address [ MAC_address | rbridge-id
rbridge-id ] ] ] | interface {<N> gigabitethernet rbridge-id/slot/port | vlan vlan_id } | dynamic [ address MAC_address |
interface {<N> gigabitethernet rbridge-id/slot/port | port-channel number | tunnel number | vlan vlan_id } | interface { <N>
gigabitethernet rbridge-id/slot/port | port-channel number | tunnel number | vlan vlan_id } | static [ address
MAC_address | interface {<N> gigabitethernet rbridge-id/slot/port | port-channel number | tunnel number | vlan vlan_id } |
vlan vlan_id | dynamic | interface | learning-mode [ rbridge-id [ rbridge-id ] ] | linecard interface | port-profile
[ address MAC_address | count | dynamic | vlan vlan_id ] | static | vlan vlan_id ]
```

### Parameters

#### address *MAC\_address*

Displays forwarding information for a 48-bit MAC address. The valid format is *H.H.H* (available in Privileged EXEC mode only).

#### aging-time

Displays the aging time.

##### conversational

Displays conversational MAC learning (CML) aging time, or is used for forwarding entries.

##### *rbridge-id* [*rbridge-id*]

Specifies the RBridge ID display.

#### authenticated

Displays all MAC addresses learnt on a particular interface that are authenticated by the authentication server (RADIUS server). It does not show the MAC addresses whose authentication is pending or denied.

#### authentication-failed

Displays all MAC addresses learnt on a particular interface whose authentication was denied by the authentication server (RADIUS server).

#### count

Displays the count of forwarding entries.

##### address *MAC\_address*

Specifies a MAC address.

##### conversational linecard *linecard\_number*

Specifies CML addresses for a line card.

##### address

Specifies a MAC address or an RBridge.

- interface
  - Specifies a physical interface or VLAN.
- <N>**gigabitethernet**
  - Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.
- rbridge-id**[*rbridge-id*]
  - Specifies all RBridge IDs or a single RBridge ID.
- slot*
  - Specifies a valid slot number.
- port*
  - Specifies a valid port number.
- port-channel** *number*
  - Specifies a port-channel number. Range is from 1 through 6144.
- tunnel** *number*
  - Specifies a tunnel. Range is from 1 through 100000.
- dynamic
  - Specifies dynamic (legacy) MAC addresses for a physical interface, port-channel, or VLAN.
- static
  - Specifies static MAC addresses for a physical interface, port-channel, or VLAN.
- learning-mode
  - Specifies the learning mode for all RBridges or one RBridge.
- linecard interface
  - Specifies a line card.
- port-profile
  - Specifies a port profile.

## Modes

Privileged EXEC mode

## Usage Guidelines

In Network OS 7.2.0, the SNMP feature learns active and local MAC addresses in a VDX cluster. For locally learned MAC addresses, the port number is displayed. For remotely learned MAC addresses, the port number is 0.

## Examples

To display the status of all MAC addresses in the table:

```
device# show mac-address-table
VlanId  Mac-address      Type      State      Ports
1       0000.0000.0001   Static    Remote     Te 1/4/5
1       0000.0000.0002   Static    Active     Te 3/0/11
1       0010.9400.0002   Dynamic   Active     Te 3/0/11
1       0011.9400.0002   Dynamic   Remote     Te 1/4/5
Total MAC addresses      : 4
```

To display a specific MAC address in the table:

```
device# show mac-address-table address 0011.2222.3333
vlanId  Mac-address      Type      State      Ports
100     0011.2222.3333   Static    Inactive    Te 1/0/1
Total MAC addresses      : 1
```

To display the aging time for a specific MAC address table:

```
device# show mac-address-table aging-time
MAC Aging-time : 300 seconds
```

To display the authenticated MAC addresses on an interface:

```
device# show mac-address-table authenticated interface tengigabitethernet 12/1/40
VlanId  Mac-address      State
=====
1       0000.1121.0102   AUTHENTICATED
1       0000.1121.0104   AUTHENTICATED
1       0000.1121.0106   AUTHENTICATED
1       0000.1121.0108   AUTHENTICATED
```

To display the MAC addresses whose authentication is denied by the authentication server:

```
device# show mac-address-table authentication-failed interface tengigabitethernet 12/1/22
VlanId  Mac-address      State
=====
1       0000.1124.0101   FAILED
1       0000.1124.0103   FAILED
1       0000.1124.0105   FAILED
1       0000.1124.0107   FAILED
1       0000.1124.0109   FAILED
1       0000.1124.010a   FAILED
```

To display the status of dynamic MAC addresses in the table:

```
device# show mac-address-table dynamic
vlanId  Mac-address      Type      State      Ports
100     0011.2222.5555   Dynamic   Inactive    Te 1/0/1
100     0011.2222.6666   Dynamic   Inactive    Te 1/0/1
Total MAC addresses      : 2
```

To display the count of all MAC addresses, including BGP EVPN addresses, in the table for a specific VLAN:

```
device# show mac-address-table count vlan 152
Dynamic Address Count : 0
Static Address Count  : 0
EVPN Address Count    : 3
Internal Address Count: 1
Total MAC addresses   : 4
```

To display the count of all MAC addresses, including BGP EVPN addresses, in the table for a specific tunnel:

```
device# show mac-address-table count interface tunnel 61441
Dynamic Address Count      : 0
Static Address Count      : 0
EVPN Address Count        : 40
Internal Address Count    : 0
Total MAC addresses       : 40
```

To display status of all MAC addresses, including BGP EVPN addresses, in the table for a specific tunnel:

```
device# show mac-address-table interface tunnel 61441
VlanId  Mac-address      Type   State   Ports
111     50eb.1aaa.30b1      EVPN   Active  Tu 61441
111     50eb.1aac.2c41      EVPN   Active  Tu 61441
Total MAC addresses      : 2
```

## History

Release version	Command history
7.1.0	This command was modified to show examples of the status of BGP EVPN MAC addresses. Also, this command was modified to display list of authenticated and non-authenticated MAC addresses.
7.2.0	This command was modified to show remote and active states.

# show mac-address-table consistency-check

Displays the operational data for MAC address-table consistency check.

## Syntax

```
show mac-address-table consistency-check
```

## Modes

Privileged EXEC mode

## Command Output

The `show mac-address-table consistency-check` command displays the following information:

Output field	Description
Consistency Check	Indicates if MAC address-table consistency check is enabled or disabled.
Consistency Check Interval	Displays the time interval, in seconds, for the MAC consistency-check trigger.

## Examples

The following example indicates that MAC-move detection is enabled for consistency check. Consistency check is configured to run once every 300 seconds.

```
device# show mac-address-table consistency-check
Consistency Check      : Enabled
Consistency Check Interval: 300 seconds
```

## History

Release version	Command history
6.0.0	This command was introduced.

## Related Commands

[mac-address-table consistency-check interval](#), [mac-address-table consistency-check suppress](#)



# show mac-address-table count evpn

Displays the count of all MAC addresses in the Layer 2 forwarding database that are distributed by means of BGP EVPN.

## Syntax

```
show mac-address-table count evpn
```

## Modes

Privileged EXEC mode

## Examples

To display the count of all MAC addresses in the Layer 2 forwarding database that are distributed by means of BGP EVPN:

```
device# show mac-address-table count evpn
EVPN Address Count: 50
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show mac-address-table evpn

Displays all MAC addresses in the Layer 2 forwarding database that are distributed by means of BGP EVPN.

## Syntax

`show mac-address-table evpn`

## Modes

Privileged EXEC mode

## Examples

To display all MAC addresses in the Layer 2 forwarding database that are distributed by means of BGP EVPN:

```
device# show mac-address-table evpn
VlanId  Mac-address      Type   State   Ports
111     50eb.1aaa.30b1   EVPN   Active  Tu 61441
111     50eb.1aac.2c41   EVPN   Active  Tu 61441
112     50eb.1aaa.30b1   EVPN   Active  Tu 61441
112     50eb.1aac.2c41   EVPN   Active  Tu 61441
Total MAC addresses      : 4
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show mac-address-table mac-move

Displays the operational data for MAC address-table MAC-move detection,

## Syntax

**show mac-address-table mac-move**

## Modes

Privileged EXEC mode

## Command Output

The **show mac-address-table mac-move** command displays the following information:

Output field	Description
Mac Move detect	Indicates whether MAC address-table MAC-move detection is enabled or disabled.
Threshold	Indicates the time interval, in seconds, for the MAC-address-move resolution trigger.
Action	Indicates the action to be taken: shutdown (including RASLog) or RASLog alone (without shutdown).
Auto-recovery	Indicates whether autorecovery is enabled or not.
Auto-recovery-time	The interval, in minutes, between MAC-move response and recovery.

## Examples

The following example indicates that MAC-move detection is enabled. MAC-address-move resolution is triggered if more than 20 MAC moves are detected within any 10-second window. The auto-recovery time is the default of 5 minutes.

```
device# show mac-address-table mac-move
Mac Move detect :      Enabled/Disabled
Threshold:             20
Action:                Raslog/shutdown
Auto-recovery:         Enabled/Disabled
Auto-recovery-time:   5
```

## History

Release version	Command history
6.0.0	This command was introduced.

# show maps dashboard

Displays the Monitoring and Alerting Policy Suite (MAPS) data collected since midnight.

## Syntax

```
show maps dashboard [ details | history ] | [ rbridge-id [ rbridge_id | all ] ]
```

## Parameters

### details

Displays detailed MAPS statistics. Displays only five rules and the top five ports.

### history

Displays a briefer summary of the dashboard information than the **all** keyword.

### rbridge-id *rbridge\_id*

Specifies an RBridge ID for which to display the MAPS statistics.

### all

Displays all of the dashboard information.

## Modes

Privileged EXEC mode

## Usage Guidelines

The output of the command is divided into five basic sections:

- 1.0 Basic information about the dashboard.
- 2.0 A status report on the current health of the switch.
- 3.1 A summary of the focus areas of the switch.
- 3.2 A list of the rules currently affecting the health of the switch.
- 3.3 A historical summary of the events on the switch over the last 24 hours.

## Examples

The following example displays all MAPS statistics since midnight for RBridge 10.

```
device# show maps dashboard rbridge-id 10
-----
                        Dashboard for RbridgeId 10
-----

1 Dashboard Information:
=====

DB start time :                Thu May 21 17:27:28 2015

2 Switch Health Report:
=====

Current Switch Policy Status: MARGINAL
Contributing Factors:
-----
*BAD_PWR (MARGINAL).

3.1 Summary Report:
=====

Category                |Today                |Last 7 days          |
-----
Port Health              |No Errors            |No Errors            |
Fru Health               |In operating range   |In operating range   |
Security Violations      |No Errors            |Out of operating range |
Switch Resource          |In operating range   |In operating range   |

3.2 Rules Affecting Health:
=====

Category(Rule Count)|RepeatCount|Rule Name                |Execution Time |Object                |
Triggered Value(Units)|          |                          |               |                      |
-----
Security Violations(|2          |defSWITCHSEC_LV_0        |05/21/15 19:26:54|Switch              |1
Violations          |          |                          |               |                      |
3)                  |          |                          |               |                      |
|                   |          |                          |               |                      |
Violations          |          |                          |               |Switch              |1
|                   |          |                          |               |                      |
Violations          |1         |defSWITCHSEC_TELNET_0    |05/21/15 19:26:54|Switch              |1
Violations          |          |                          |               |                      |

3.3 History Data:
=====

Stats(Units)          Current          --/--/--          --/--/--          --/--/--
--/--/--              --/--/--        --/--/--          --/--/--          --/--/--
Port (val)            Port (val)      Port (val)         Port (val)         Port (val)
-----
CRCALN (CRCs)         -               -                  -                  -
RX_ABN_FRAME (Errors)-               -                  -                  -
RX_SYM_ERR (Errors)  -               -                  -                  -
RX_IFG (IFGs)        -               -                  -                  -
-                    -               -                  -                  -
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show maps group

Displays the Monitoring and Alerting Policy Suite (MAPS) groups configured on the device.

## Syntax

```
show maps group [ rbridge-id value]
```

## Parameters

**rbridge-id** *value*

Specifies an RBridge ID. The range of valid values is from 1 through 239.

## Modes

Privileged EXEC mode

## Command Output

The **show maps group** command displays the following information:

Output field	Description
Group	The group name.
Type	The type of members in the group.
Members	Current members of the group, if any

## Examples

Typical command example.

```
device# show maps group rbridge 148
-----
                Groups for RbridgeId 148
-----
  Group          Type          Members
-----
  ALL_PORTS      Eth Port
  ALL_TS         Sensor
  ALL_FAN        Fan
  ALL_PS         Power
  ALL_SFP        Sfp
  ALL_WWN        WWN
  ALL_1G_SFP     Sfp
  ALL_1GSR_SFP  Sfp
  ...
  ALL_ETH_PORTS  Eth Port
  ALL_iSCSI_PORTS Eth Port
  ...
There are 2 user defined groups in rbridge 148

  test6         sfp          148/0/1-5,10,15-20
  test7         interface    148/1/11-15,17,20-25
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show maps policy

Displays the Monitoring and Alerting Policy Suite (MAPS) policies enforced on the device.

## Syntax

```
show maps policy [summary | detail | name policy_name | rbridge-id rbridge_id | all ]
```

## Parameters

### summary

Displays a summary of the MAPS statistics.

### detail

Displays a detailed output of the MAPS statistics.

### name *policy\_name*

Displays all the rules in the specified policy.

### rbridge-id *rbridge\_id*

Specifies an RBridge ID for which to display the MAPS statistics.

## Modes

Privileged EXEC mode

## Examples

Typical command example.

```
device# #show maps policy summary rbridge-id 5

      Policy Name                Number of Rules
-----
dflt_aggressive_policy          :                196
dflt_conservative_policy        :                198
dflt_moderate_policy            :                198
Active Policy is 'dflt_conservative_policy'.
```

Typical command example.

```
device# show maps policy name dflt_conservative_policy rbridge-id 1
```

```
-----
                        dflt_conservative_policy Rules (NOT ENABLED)
-----
Action                Rules List                Condition
-----
                        defALL_ETH_PORTS_RX_SYM_ERR_0          RASLOG,SNMP,EMAIL
RX_SYM_ERRALL_ETH_PORTS(MIN>0)
                        defALL_iSCSI_PORTS_RX_SYM_ERR_0          RASLOG,SNMP,EMAIL
RX_SYM_ERRALL_iSCSI_PORTS(MIN>0)
                        defALL_NAS_PORTS_RX_SYM_ERR_0           RASLOG,SNMP,EMAIL
RX_SYM_ERRALL_NAS_PORTS(MIN>0)
```

### Examples

```
device# show maps policy detail
```

```
-----
                        Policy details for RbridgeId 148
-----
                        dflt_aggressive_policy (Predefined policy with 125 rules)
-----
Rules List                Action                Condition
-----
                        defALL_ETH_PORTS_CRCALN_12          RASLOG,SNMP,EMAIL          CRCALN(ALL_ETH_PORTS(MIN>12))
...
                        defCHASSISETH_MGMT_PORT_STATE_UP          RASLOG,SNMP,EMAIL
CHASSIS(ETH_MGMT_PORT_STATE≠UP)
There are 3 user defined policies
policyName myPolicy1 on rbridge 148 has 2 rules
myRule1                EMAIL,RASLOG          TEMP test8(HOUR<=35)
myRule2                EMAIL,FENCE          CRCALN test6(HOUR>103)
policyName myPolicy2 on rbridge 148 has 1 rules
myRule2                SNMP,EMAIL          CRCALN test6(HOUR>103)
policyName myPolicy3 on rbridge 148 has 1 rules
myRule2                SNMP                CRCALN test6(HOUR>103)
```

### History

Release version	Command history
6.0.1	This command was introduced.

# show media

Displays the SFP information for all the interfaces present on a switch.

## Syntax

```
show media
```

## Modes

Privileged EXEC mode

## Usage Guidelines

The command output will be several pages long.

The TX Power Field in the **show media** command is not supported by the 40-Gbps optics.

## Examples

To display all SFP information:

```
switch# show media
Interface Ten Gigabit Ethernet 0/1
  Identifier      3      SFP
  Connector       7      LC
  Transceiver     0000000000000010 10_GB/s
  Name            id
  Encoding        6
  Baud Rate       103 (units 100 megabaud)
  Length 9u       0      (units km)
  Length 9u       0      (units 100 meters)
  Length 50u      8      (units 10 meters)
  Length 62.5u   3      (units 10 meters)
  Length Cu       0      (units 1 meter)
  Vendor Name     Extreme
  Vendor OUI      42:52:4f
  Vendor PN       57-0000075-01
  Vendor Rev      A
  Wavelength      850 (units nm)
  Options         001a Loss_of_Sig,Tx_Fault,Tx_Disable
  BR Max          0
  BR Min          0
  Serial No       AAA108454100431
  Date Code       081108
  Optical Monitor yes
  Temperature     44 Centigrade
  Voltage         3246.8 (Volts)
  Current         0.002 (mAmps)
  TX Power        0.1 (uWatts)
  RX Power        0.1 (uWatts)
(Output truncated)
```

## Related Commands

[show media interface](#), [show media linecard](#)

# show media interface

Displays the SFP information for a specific interface.

## Syntax

```
show media interface [ <N>gigabitethernet rbridge-id/slot/port | fibrechannel rbridge-id/slot/port ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### fibrechannel rbridge-id/slot/port

Specifies a valid external 1-gigabit FibreChannel interface.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **fibrechannel** *rbridge-id/slot/port* parameter is used only on Extreme VDX 6740, VDX 2741, and VDX 2746 switches.

## Examples

To display SPF information for a 1-gigabit Ethernet interface:

```
switch# show media interface gigabitethernet 1/0/1

Interface          Gigabit Ethernet 0/1
Identifier         2      On-board
Connector         34     CAT-5 copper cable
Transceiver       1000   BASE-T Gigabit Ethernet
Name              cu
Encoding          5      IEEE 802.3ab
Length            max 100 m
Copper Speed      1GB/s Fixed
Copper Duplex     Full Duplex
Sync status       Valid/No
Vendor Name       Broadcom
Vendor OUI        00:1B:E9
Vendor model      02:0F
Vendor Rev        01
Options           001a Remote fault/Jabber detect/copper link up
Temperature threshold/val 55 Centigrade
Voltage threshold/val    3289.9 (mVolts)
```

To display SFP information for a 10-gigabit Ethernet interface:

```
switch# show media interface tengigabitethernet 5/0/1

Interface Ten Gigabit Ethernet 5/0/1
Identifier 3      SFP
Connector 7      LC
Transceiver 0000000000000010 10_GB/s
Name      id
Encoding 6
Baud Rate 103 (units 100 megabaud)
Length 9u 0 (units km)
Length 9u 0 (units 100 meters)
Length 50u 8 (units 10 meters)
Length 62.5u 3 (units 10 meters)
Length Cu 0 (units 1 meter)
Vendor Name Extreme
Vendor OUI 00:05:1E
Vendor PN 57-0000075-01
Vendor Rev A
Wavelength 850 (units nm)
Options 001a Loss_of_Sig,Tx_Fault,Tx_Disable
BR Max 0
BR Min 0
Serial No AAA108454100431
Date Code 081108
Temperature 44 Centigrade
Voltage 3246.8 (Volts)
Current 0.002 (mAmps)
TX Power 0.1 (uWatts)
RX Power 0.1 (uWatts)
```

To display SFP information for a Fibre Channel interface:

```
switch# show media interface fibrechannel 66/0/1
```

```

Interface          FibreChannel 66/0/1
Identifier         3      SFP
Connector         7      LC
Transceiver       540c404040000000 200,400,800_MB/s M5,M6 sw Short_dist
Encoding          1      8B10B
Baud Rate         85     (units 100 megabaud)
Length 9u         0      (units km)
Length 9u         0      (units 100 meters)
Length 50u        5      (units 10 meters)
Length 62.5u     2      (units 10 meters)
Length Cu         0      (units 1 meter)
Vendor Name       Extreme
Vendor OUI        00:05:1e
Vendor PN         57-1000012-01
Vendor Rev        A
Wavelength        850    (units nm)
Options           003a Loss_of_Sig,Tx_Fault,Tx_Disable
BR Max            0
BR Min            0
Serial No         UAF110170000VP1
Date Code         100422
DD Type           0x68
Enh Options       0xfa
Status/Ctrl       0x82
Alarm flags[0,1] 0x5, 0x40
Warn Flags[0,1]  0x5, 0x40

                                Alarm          Warn
                                low          high        low          high
Temperature 28      Centigrade    -10         90          -5           85
Voltage      3331.4  mVolts      2900.0     3700.0     3000.0     3600.0
Current      0.310     mAmps       1.000      17.000     2.000      14.000
TX Power     -21.7     dBm (6.8 uW) 125.9 uW   631.0 uW   158.5 uW   562.3 uW
RX Power     -inf      dBm (0.0 uW) 10.0 uW    1258.9 uW  15.8 uW    1000.0 uW

```

## Related Commands

[show media](#), [show media linecard](#)

# show media linecard

Displays the SFP information for a specified line card.

## Syntax

```
show media linecard number
```

## Parameters

*number*

Numeric identifier for the line card.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display a summary of small form-factor pluggable (SFP) and Quad SFP media information for each interface on the specified module.

This command is supported only on the local RBridge.

## Examples

To display the SFP media information for an LC48X10G line card in slot 2:

```
switch# show media linecard 2

Interface      Ten Gigabit Ethernet 1/2/1
Identifier     3      SFP
Connector     33     Copper Pigtail
Transceiver    d580884104000002 10_GB/s TW Short_dist
Name          cu
Encoding      0
Baud Rate     103 (units 100 megabaud)
Length 9u     0 (units km)
Length 9u     0 (units 100 meters)
Length 50u    0 (units 10 meters)
Length 62.5u 0 (units 10 meters)
Length Cu     1 (units 1 meter)
Vendor Name    Extreme
Vendor OUI    00:05:1e
Vendor PN     58-1000026-01
Vendor Rev    A
Wavelength    3072(units nm)
Options       0012
BR Max        0
BR Min        0
Serial No     CAMB110100607EW
Date Code     110111
Optical Monitor No
Temperature   N/A
Voltage       N/A
Current       N/A
TX Power      N/A
RX Power      N/A
(Output truncated)
```

To display the Quad SFP media information for an LC12X40G line card in slot 3:

```
switch# show media linecard 3

Interface      fortygigabitethernet 1/3/2
Identifier     13     QSFP
Connector     12
Transceiver    0000000000000004 40_GB/s Short_dist
Name          sw
Encoding      5      IEEE 802.3ab
Baud Rate     103 (units 100 megabaud)
Length 9u     0 (units km)
Length 9u     50 (units 100 meters)
Length 50u    0 (units 10 meters)
Length 62.5u 0 (units 10 meters)
Length Cu     0 (units 1 meter)
Vendor Name    5ROCADE
Vendor OUI    00:05:1e
Vendor PN     57-1000128-01
Vendor Rev    A
Wavelength    17000(units nm)
Options       0000
BR Max        15
BR Min        222
Serial No     LTA111421000923
Date Code     111022
Optical Monitor yes
Temperature   31 Centigrade
Voltage       3313.2 (mVolts)
Current       7.204 (mAmps)
TX Power      N/A
RX Power      0.0 (uWatts)
```



## Related Commands

[linecard](#), [show slots](#)

# show media optical-monitoring interface

Displays the configuration values and environmental information for optical interfaces.

## Syntax

```
show media optical-monitoring interface [ <N>gigabitethernet rbridge-id/slot/port | fibrechannel rbridge-id/slot/port ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### fibrechannel *rbridge-id/slot/port*

Specifies a valid external 1-gigabit FibreChannel interface.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

Sample output for all interfaces.

```
device# show media optical-monitoring interface
N/A - Not Available.
N/S - Optical-monitoring Not Supported.
Port          Module      Supply      Channel      Frequency  Wavelength  Bias          Channel
              Temperature Voltage      TX Power      Error      Error      Current      RX Power
              ( C )        ( mVolts )  ( uWatts )   ( GHz )   ( nm )    ( mAmps )    ( uWatts )
=====
Fo 1/0/52     31          3306.4      N/A          37.216    2.7         34.424       3.1
              33.804    4.0         33.348       1.0

Te 1/0/1      36          3293.8      694.4        0.0       0.000       38.554       747.7

Te 1/0/48     N/S

Te 2/0/1      33          3315.6      687.5        0.0       0.000       37.192       910.4

Te 2/0/48     N/S
```

Sample output for a single interface.

```
device# show media optical-monitoring interface tengigabitethernet 2/0/1
N/A - Not Available.
N/S - Optical-monitoring Not Supported.
Port          Module      Supply      Channel      Frequency  Wavelength  Bias          Channel
              Temperature Voltage      TX Power      Error      Error      Current      RX Power
              ( C )        ( mVolts )  ( uWatts )   ( GHz )   ( nm )    ( mAmps )    ( uWatts )
=====
Te 2/0/1      33          3314.9      689.8        0.0       0.000       37.014       912.1
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show media tunable-optic-sfpp

Displays the channels on which the tunable optic interfaces are currently operating.

## Syntax

```
show media tunable-optic-sfpp [ channel channel_number]
```

## Parameters

**channel** *channel\_number*

The channel number to display. The range of valid values is from 0 through 102.

## Modes

Privileged EXEC mode

## Command Output

The **show media tunable-optic-sfpp** command displays the following information:

Output field	Description
Channel	The number assigned to the channel.
Wavelength	The wavelength on which the optic interface is operating.

## Examples

Sample output for a single channel.

```
device# show media tunable-optic-sfpp channel 2
command is show-media-tunable-optic-sfpp-channel-2.
  Channel  Wavelength
  =====  =====
  2         1568.36
```

Sample output for all channels.

```
device# show media tunable-optic-sfpp
command is show-media-tunable-optic-sfpp.
Channel    Wavelength
=====
1          1568.77
2          1568.36
3          1567.95
4          1567.54
5          1567.13
6          1566.72
7          1566.31
8          1565.90
9          1565.50
10         1565.09
11         1564.68
12         1564.27
13         1563.86
14         1563.45
15         1563.05
16         1562.64
17         1562.23
18         1561.83
19         1561.42
20         1561.01
(Output truncated for brevity.)
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show mgmt-ip-service

Displays the status of management IP services.

## Syntax

```
show mgmt-ip-service [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

To display the status of management IP services on all platforms in a cluster:

```
device# show mgmt-ip-service
```

To display the status of management IP services on RBridge 10:

```
device# show mgmt-ip-service rbridge-id 10
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show mm

Displays information about the Management Modules present in the chassis.

## Syntax

```
show mm [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID. The range of valid values is from 1 through 239.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic tests (POST1, POST2) are not running on the management modules.

## Command Output

The **show mm** command displays the following information:

Output field	Description
Slot	Displays the slot number. Slots for management modules are M1 and M2.
Type	Displays the line card type. The management module type is MM.
Description	Module description
ID	Displays the module ID. The ID for the management module is 112.
Status	Displays the status of the module as one of the following: <ul style="list-style-type: none"> <li>• VACANT - The slot is empty.</li> <li>• POWERED-OFF - The module is present in the slot but is powered off.</li> <li>• POWERING UP - The module is present and powering on.</li> <li>• LOADING - The module is present, powered on, and loading the initial configuration.</li> <li>• INITIALIZING - The module is present, powered on, and initializing hardware components.</li> <li>• ENABLED - The module is on and fully enabled.</li> <li>• DISABLED - The module is powered on but disabled.</li> <li>• FAULTY - The module is faulty because an error was detected.</li> <li>• UNKNOWN - The module is inserted but its state cannot be determined.</li> </ul>

## Examples

The following example displays the management modules present in an Extreme VDX 8770-4 chassis.

```
device# show mm
```

Slot	Type	Description	ID	Status
M1	MM	Management Module	112	ENABLED
M2				VACANT

The following example displays the management modules present on RBridge 80.

```
device# mm rbridge-id 80
Rbridge-id 80:
```

Slot	Type	Description	ID	Status
M1	MM	Management Module	112	ENABLED
M2	MM	Management Module	112	ENABLED

## History

Release version	Command history
6.0.0	This command was modified to support the <b>rbridge-id</b> parameter.

## Related Commands

[show linecard](#), [show sfm](#), [show slots](#)



# show monitor

Displays the monitoring information for all Port Mirroring sessions or for a single session.

## Syntax

```
show monitor [ session session_number ]
```

## Parameters

**session** *session\_number*

Specifies a session identification number. Valid values range from 0 through 511.

## Modes

Privileged EXEC mode

## Examples

To display monitoring information for all Port Mirroring sessions:

```
switch# show monitor

Session           :1
Type              :Remote source session
Description       :Test monitor session
State             :Enabled
Source interface  :Te 1/0/10 (Up)
Destination interface :Vlan x
Direction        :Rx
```

## Related Commands

[monitor session](#)

# show name-server brief

Displays brief entries of local name server (NS) information about devices connected to a switch.

## Syntax

```
show name-server brief [ rbridge-id rbridge-id ]
```

## Command Default

If no RBridge ID is specified, brief entries for all devices in the fabric are displayed.

## Parameters

**rbridge-id** *rbridge-id*  
Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Brief output lists only the Fibre Channel address (PID) of each device.

### NOTE

If no information is available for the switch, the command displays the message: "0 entries."

## Examples

```
switch# show name-server brief
```

Example output, with no hidden entries, follows; note that there are two spaces preceding each PID:

```
481000 481040 471000 471040  
4 entries
```

Hidden entries are preceded by a single space, followed by an asterisk [\*]. Output from RBridge 0x47 perspective (PID 471080 hidden, PID 481000 removed) follows:

```
471040 *471080 481040  
2 entries  
* 1 hidden entries due to duplicate WWN
```

## Related Commands

[show name-server detail](#), [show name-server nodefind](#)

# show name-server detail

Displays local name server (NS) information about devices connected to a switch.

## Syntax

```
show name-server detail [ rbridge-id rbridge-id ]
```

## Command Default

If no RBridge ID is specified, detailed entries for all devices in the fabric are displayed.

## Parameters

**rbridge-id** *rbridge-id*  
Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Command Output

The **show name-server detail** command displays the following information:

Output field	Description
PID	Fibre Channel address of the device in 6-digit hexadecimal format.
Port Name	Worldwide port name (WWPN) of the device.
Node Name	Worldwide node name (WWNN) of the device.
SCR	Indicates the State Change Registration of the device, which affects RSCN behavior. The value can be one of the following: <ul style="list-style-type: none"> <li>0: No registration</li> <li>1: Fabric detected RSCN</li> <li>2: Port detected RSCN</li> <li>3: Both Fabric and Port detected RSCN</li> </ul> <p><b>NOTE</b> This information is only available for devices that are connected to the local RBridge on which the command is executed.</p>
FC4s	Fibre Channel FC4 type of the device represented as an ASCII string.
PortSymb	User defined name for this port (ASCII string).
Fabric Port Name	Fabric port name (worldwide name format). This is the F_Port worldwide name to which the N_Port connects.
Permanent Port Name	Physical Nx_Port worldwide name.
Device Type	Type and role of the device, where the device type is either "Physical", "Virtual", "NPIV", or "iSCSI". The role is either "Initiator", "Target", or "Initiator + Target". If the device role is not

Output field	Description
	registered, the display indicates "unknown". If the device registers a type that is not one of the aforementioned values, then the type is listed as "undefined".
Interface	Interface information for the port. For FCoE devices this information is shown as: Fcoe vlan/rbridge-id/FCoE port.
Physical Interface	Physical interface information for the port. For FCoE devices this information is shown as: Te rbridge-id/slot/port Where Te = Ten gigabit Ethernet.
Share Area	The state of the Extreme shared area addressing method. If "Yes" then the port uses shared area addressing.
Redirect	Indicates whether or not the device is involved in Extreme Frame Redirection. If "Yes" then the device is involved in Frame Redirection zoning.

If no information is available for the switch, the command displays the message: "total number of 0 entries."

For each detail entry displayed, if the device has been flagged as having a duplicate Port WWN, the following text appears after the entry:

```
*** Duplicate Port WWN entry - PID: <PID>
*** Please see RAS log entry NS-1012 pertaining to this device
```

## Examples

To view name server entries for all devices connected to a switch:

```
switch# show name-server detail

PID: 012100
Port Name: 10:00:00:05:1E:ED:95:38
Node Name: 20:00:00:05:1E:ED:95:38
SCR: 3
FC4s: FCP
PortSymb: [27] "Extreme-1020|2.3.0.0|localhost.localdomain|Red Hat
Enterprise Linux Server release 5.5"
NodeSymb: NULL
Fabric Port Name: 20:21:00:05:1E:CD:79:7A
Permanent Port Name: 10:00:00:05:1E:ED:95:38
Device type: Physical Initiator
Interface: Fcoe 1/1/9
Physical Interface: Te 1/0/9
Share Area: No
Redirect: No
```

## Related Commands

[show name-server brief](#), [show name-server nodefind](#)

# show name-server nodefind

Displays the local name server (NS) information for a specific device.

## Syntax

```
show name-server nodefind { PID pid | WWN wwn }
```

## Parameters

**PID** *pid*

Specifies the Fibre Channel address of the device to search for.

**WWN** *wwn*

Specifies the World Wide Name (WWN) of the device to search for.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display local name server (NS) information about a specific detailed entry.

The lookup is performed using either the Fibre Channel address (Port ID, or PID), or the WWN of the device. If no information is available for the device, the command displays one of the following messages:

- For PID lookup: Device with PID of *pid* does not exist
- For WWN lookup: Device with WWN of *wwn* does not exist

Refer to **show name-server detail** for descriptions of the displayed information.

## Examples

To view name server information for a device specified by PID:

```
switch# show name-server nodefind pid 0x012100

PID: 012100
Port Name: 10:00:00:05:1E:ED:95:38
Node Name: 20:00:00:05:1E:ED:95:38
SCR: 3
FC4s: FCP
PortSymb: [27] "Extreme-1020|2.3.0.0|localhost.localdomain|Red Hat
Enterprise Linux Server release 5.5"
NodeSymb: NULL
Fabric Port Name: 20:21:00:05:1E:CD:79:7A
Permanent Port Name: 10:00:00:05:1E:ED:95:38
Device type: Physical Initiator
Interface: Fcoe 1/1/9
Physical Interface: Te 1/0/9
Share Area: No
Redirect: No
```

show name-server nodefind

## Related Commands

[show name-server brief](#), [show name-server detail](#)

# show name-server zonemember

Displays the local name server (NS) zoning information.

## Syntax

```
show name-server zonemember { PID pid | WWN wwn }
```

## Parameters

**PID** *pid*

Specifies the Fibre Channel address of the device.

**WWN** *wwn*

Specifies the World Wide Name (WWN) of the device.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display detailed local name server (NS) information about all devices zoned with the specified device. The lookup is performed using either the Fibre Channel address (Port ID, or PID), or the WWN of the device.

If no information is available for the device, the command displays one of the following messages:

- For PID lookup: Device with PID of *pid* does not exist
- For WWN lookup: Device with WWN of *wwn* does not exist

If the specified device has been flagged as having a duplicate Port WWN, the following text is displayed:

```
Device with duplicate WWN removed from Name Server database: <device detail entry displayed here>  
Please review RAS log for NS-1012 entries in order to obtain details about the conflicting device pair.
```

## Examples

To view name server information for a device specified by PID:

```
switch# show name-server zonemember pid 0x010500

PID: 010500
Port Name: 10:00:00:05:1F:ED:95:38
Node Name: 20:00:00:05:1F:ED:95:38
SCR: 3
FC4s: FCP
PortSymb: [27] "Extreme-1020|2.3.0.0|localhost.localdomain|Red Hat
Enterprise Linux Server release 5.5"
NodeSymb: NULL
Fabric Port Name: 20:21:00:05:1F:ED:79:7A
Permanent Port Name: 10:00:00:05:1F:ED:95:38
Device type: Physical Initiator
Interface: Fcoe 1/1/5
Physical Interface: Te 1/0/5
Share Area: No
Redirect: No
PID: 010600
Port Name: 10:00:00:05:1F:CD:95:38
Node Name: 20:00:00:05:1F:CD:95:38
SCR: 3
FC4s: FCP
PortSymb: [27] "Extreme-1020|2.3.0.0|localhost.localdomain|Red Hat
Enterprise Linux Server release 5.5"
NodeSymb: NULL
Fabric Port Name: 20:21:00:05:1E:CD:79:7A
Permanent Port Name: 10:00:00:05:1E:CD:95:38
Device type: Physical Initiator
Interface: Fcoe 1/1/6
Physical Interface: Te 1/0/6
Share Area: No
Redirect: No
```

## Related Commands

[show name-server brief](#), [show name-server detail](#), [show name-server nodefind](#)



# show nas statistics

Displays automatic network attached storage (Auto NAS) statistics.

## Syntax

```
show nas statistics all | server-ip ip_addr/prefix [ vlan VLAN_id | vrf VRF_name ] [ rbridge-id rbridge-id ]
```

## Parameters

### all

Shows all gathered statistics.

### server-ip

IP address to show Auto NAS statistics for.

*ip\_addr/prefix*

IPv4 address/prefix of a specified **Auto** NAS port.

**vlan** *VLAN\_id*

Specifies which VLAN interface to display the statistics for.

**vrf** *VRF\_name*

Specifies which VRF interface to display the statistics for.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on Extreme VDX 8770-4, VDX 8770-8, VDX 6740, and VDX 6740T devices.

# show netconf client-capabilities

Displays the client capabilities associated with each NETCONF session.

## Syntax

```
show netconf client-capabilities
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display client capabilities for all active NETCONF sessions. It always displays the session-ID, login name of the user of the client session, the host IP address, and the time the user logged on. The application vendor name, application product name and version number, and the identity of the client are also returned if these values are advertised by the client as capabilities in the <hello> message to the server at the start of the session.

## Examples

```
device# show netconf client-capabilities
```

```
Session Id   : 10
User name    : root
Vendor       : Extreme
Product      : Network Advisor
Version      : 9.1.0 Build 123
Client user  : admin-user
Host IP      : 10.24.65.8
Login time   : 2011-08-18T08:54:24Z
Session Id   : 11
User name    : root
Vendor       : Not Available
Product      : Not Available
Version      : Not Available
Client user  : Not Available
Host IP      : 10.24.65.8
```

## Related Commands

[show netconf-state capabilities](#), [show netconf-state statistics](#)

# show netconf-state capabilities

Displays NETCONF server capabilities.

## Syntax

```
show netconf-state capabilities
```

## Modes

Privileged EXEC mode

## Examples

```
device# show netconf-state capabilities

netconf-state capabilities capability urn:ietf:params:netconf:base:1.0
netconf-state capabilities capability urn:ietf:params:netconf:capability:writable-running:1.0
netconf-state capabilities capability urn:ietf:params:netconf:capability:startup:1.0
netconf-state capabilities capability urn:ietf:params:netconf:capability:xpath:1.0
netconf-state capabilities capability urn:ietf:params:netconf:capability:validate:1.0
netconf-state capabilities capability http://tail-f.com/ns/netconf/actions/1.0
netconf-state capabilities capability http://tail-f.com/ns/aaa/1.1?revision=2010-06-17&module=tailf-aaa
netconf-state capabilities capability urn:.com:mgmt:extreme-aaa?revision=2010-10-21&module=extreme-aaa
(Output truncated)
```

# show netconf-state datastores

Displays the NETCONF datastores that are present on the NETCONF server along with related locking information.

## Syntax

`show netconf-state datastores`

## Modes

Privileged EXEC mode

## Examples

```
switch# show netconf-state datastores
```

NAME	LOCKED		LOCK ID	LOCKED		SELECT	LOCKED
	BY SESSION	TIME		BY SESSION	TIME		
running	-	-					
startup	-	-					

# show netconf-state schemas

Displays the data models supported by the NETCONF server.

## Syntax

```
show netconf-state schemas
```

## Modes

Privileged EXEC mode

# show netconf-state sessions

Displays information about currently active NETCONF sessions.

## Syntax

**show netconf-state sessions**

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the following information about each active NETCONF session:

- Transport used by the session
- Login name of the user
- Client IP address
- The time the user logged in

This command also provides a summary of RPC error counts and notifications.

## Examples

```
switch# show netconf-state sessions

etconf-state sessions session 6
transport cli-console
username admin
source-host 127.0.0.1
login-time 2011-09-05T11:29:31Z
netconf-state sessions session 9
transport netconf-ssh
username root
source-host 172.21.132.67
login-time 2011-09-05T11:50:33Z
in-rpcs 0
in-bad-rpcs 0
out-rpc-errors 0
out-notifications 0
```

# show netconf-state statistics

Displays NETCONF server statistics.

## Syntax

```
show netconf-state statistics
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display statistics related to the NETCONF server, including counts of the following entities:

- Start time of the NETCONF server
- Erroneous <hello> elements received
- Client sessions begun
- Dropped sessions
- Remote procedure calls (RPCs) received
- Erroneous RPCs received
- RPC errors returned to clients
- Notifications sent

## Examples

```
switch# show netconf-state statistics

netconf-state statistics netconf-start-time 2012-04-27T09:12:09Z
netconf-state statistics in-bad-hellos 0
netconf-state statistics in-sessions 3
netconf-state statistics dropped-sessions 0
netconf-state statistics in-rpcs 4
netconf-state statistics in-bad-rpcs 0
netconf-state statistics out-rpc-errors 0
netconf-state statistics out-notifications 0
```

## Related Commands

[show netconf-state capabilities](#), [show netconf-state datastores](#), [show netconf-state schemas](#), [show netconf-state sessions](#)

show notification stream

# show notification stream

Displays notifications about the event stream.

## Syntax

`show notification stream`

## Modes

Privileged EXEC mode



# show nsx-controller

Displays connection status and statistics for the NSX Controller.

## Syntax

```
show nsx-controller [ brief | client-cert | name name ]
```

## Parameters

### brief

Shows a brief listing of NSX Controller connections.

### client-cert

Displays the public certificate used for the NSX Controller connection.

### name *name*

Displays the name of the NSX Controller profile that has been configured.

## Modes

Privileged EXEC mode

## Command Output

The **show nsx-controller** command displays the following information:

State	Meaning
Connected	Connection is up and operational.
Not activated	User has shut down the connection.
Connection in progress	Switch is attempting to connect to the NSX controller.
Connection lost	Disconnected by peer, or network reachability has been lost. The switch will automatically attempt to connect after the configured amount of reconnect-interval seconds.
Connection dead	Switch could not connect to the NSX controller after the maximum number of reconnect attempts. The user can restart the connection via the "nsx-controller reconnect" command (Privileged EXEC mode) or via the "no activate" and "activate" commands in NSX Controller configuration mode.

## Examples

To show the status of the NSX Controller:

```
device# show nsx-controller
NSX controller cluster "yy"
Seed IP address 192.168.0.13, port 6632, method SSL
Reconnect interval 10 secs, Max retries unlimited
Admin state up, Number of connections 1
Number of tunnels 2, Number of MACs 4
Connection details:
  ID fb580822-b185-4068-8c9b-f15a800b4eea, Connected
  IP address 192.168.0.13, port 6632, method SSL
  Reconnect interval 10000 millis, Number of retries 0
  Local IP address 192.168.1.1, Vrf mgmt-vrf (id 0)
  Last connect time: Wed Jan 29 16:33:48 2014
  Last disconnect time: Wed Jan 29 16:33:48 2014
  Number of RPCs 63, echo 0, monitor notifications 25
```

To show a brief listing of NSX Controller connections:

```
device# show nsx-controller brief
Controller name      IP address      Port Type Connection state
=====
yy                  192.168.0.13   6632 SSL   Connected
```

To display the public certificate used for the NSX Controller connection:

```
device# show nsx-controller client-cert
-----BEGIN CERTIFICATE-----
MIIC2jCCAcICAQEwdQYJKoZIhvcNAQEFBQAwMzELMAkGA1UEBhMCQ0ExEDAOBgNV
BAoTB0Jyb2NhZGUxMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYy
Fw0xNTAxMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYyMjYy
MRiWEAYDVQQDEwlsb2NhbGhvc3QwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQCtVSTo/ZYpA591cuSGoNRiT7mPHUzn5SUyTTM6J4Z1FErYtD5iLmjZbiU4
hUd45tnSYpGstx4oArk1AobAdj1KS/Y4WuVXgQWSqQf4mLEnO5ONsaZHgt+I/TV
SL4DWqfZ/SMOYpDPW326iN6I9JIOMctcDPNm49pmroAZkePxClzuAh5LAKYIGsga
1/5gGWX2GkT0Jv5inljz43rsNpVUzylb+wTrhUbWlAFx6y6wZtAdNWz8mpoguV8E
WB7W4wo1tqyAu0X80kGocwnyRmrG/eu4PmTkBxp0QnsHfkEtLlnbu3Nt9l6v8gKn
/Omi+ts22+2jdH9OzWMuSVovxt5pAgMBAAEwdQYJKoZIhvcNAQEFBQADggEBABj2
rjDhCiByiw165SODh1Fy5+z8Pi/m4aCA1NH1yI9EteRC7nbYs94wu6DuJ5LaET3l
JWtKjY0aZ2Um0Sg9l13aG9+kkaVtn3oMgAre7/pRuxxssId7PuLibYqfz1zuwtwa
wVbtsrxUwZYW55mFOI7+ACMQKq3WUUb8S14vrNq+gB49kPJAQSYaygHZ+FdPYd01
j7B2L495jaXBtkttz/hai5BGqKwnfx1SqH0pI+RLrEvJrUHbwIMUNAcBODRZqxnX
0WmnxW5IiYnvyrZAx6AH3EdCWjkMXA3/D8VQ/eDoYNVa65um43EsHRiPSjg/AnrO
dQD04meBm7uFdgS4Gf0=
-----END CERTIFICATE-----
```

## History

Release version	Command history
7.0.0	This command was modified to correct the syntax and update the example.

# show ntp status

Displays the current active NTP server IP address or LOCL (for local switch time when no NTP servers were configured or no reachable NTP servers are available).

## Syntax

```
show ntp status [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the active NTP server. If an NTP server is not configured, the command output displays the server as "LOCL". Otherwise, the command displays the NTP server IP address.

If the RBridge ID is not provided, status results default to the local switch (LOCL). If **rbridge-id all** is specified, the command displays the status for all switches in the cluster.

## Examples

To show the local switch NTP status when an NTP server is not configured:

```
switch# show ntp status
rbridge-id 1: active ntp server is LOCL
```

To show the configured NTP server:

```
switch# show ntp status
active ntp server is 10.31.2.81
```

## Related Commands

[ntp server](#)

# show openflow

Displays the OpenFlow configuration at the global level.

## Syntax

```
show openflow [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### **all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

It includes the asynchronous messages sent to the controller.

## Examples

```

device# show openflow

Logical Instance:          1
Administrative Status:    ENABLED
Datapath-ID:              1000533e5c93f
Number of Controllers:    1

Controller 1:
Controller:                Passive
Controller Type:           OFV130
Connection Mode:          PASSIVE
Listening Address:         127.0.0.1
Connection Port:          6633
Source IP used:           NA
Connection Status:        TCP_LISTENING
Role:                      EQUAL
Asynchronous Configuration:
    Packet-in (no-match | action)
    Port-status (add | delete | modify)
    Port-status (hard-timeout | delete | grp-delete)

Controller 2:
Controller:                BVC
Controller Type:           OFV130
Connection Mode:          ACTIVE
Listening Address:         10.24.82.10
Connection Port:          6633
Source IP used:           NA
Connection Status:        OPENF_ESTABLISHED
Role:                      EQUAL
Asynchronous Configuration:
    Packet-in (no-match | action)
    Port-status (add | delete | modify)
    Port-status (hard-timeout | delete | grp-delete)

Match Capability:
L2:                        Port, Source MAC, Destination MAC, Ether type, Vlan, Vlan PCP
L3:                        Port, Vlan, Ether type, Source IP, Destination IP, IP Protocol, IP
TOS, TCP/UDP Src Port, TCP/UDP Dst Port, ICMP Type, ICMP Code

Openflow Enabled Ports:    Te 12/0/1, Te 12/0/2, Te 12/0/3

Default action:            SEND TO CONTROLLER
Maximum number of L2 flows allowed: 1024
Maximum number of L3 flows allowed: 1024
Active flow:               2048

device#

```

**TABLE 18** Output fields for the show openflow command

Field	Description
Logical Instance	Indicates the logical instance of OpenFlow controller on the device.
Administrative Status	Indicates the administrative status of OpenFlow on the device.
Data path ID	Displays the data path ID assigned to the device.
Number of Controllers	Lists the number of controller connections configured on the device.
Controller Type	Indicates the OpenFlow protocol version that is supported on the device.
Connection mode	Indicates the mode of the controller connection configured. You can configure active or passive connection to controllers. An active connection is initiated by the device. In a passive connection, the device is in the listening mode, and accepts requests from controllers. If the optional controller address is not specified, any controller can establish a connection with the device in the passive mode. If there is

**TABLE 18** Output fields for the show openflow command (continued)

Field	Description
	an address, only that IP address can connect to the device in passive mode.
Listening address	Indicates the address of the specified controller. For passive, it is local and active is remote address.
Connection port	Indicates the TCP port that is used for connection to the controller. By default, port 6633 is used.
Connection status	Indicates the status of the specified controller.
Role	Indicates the role of the specified controller.
Asynchronous configuration	Asynchronous messages sent to the specified controller.
Match capability	Specifies the matching rules supported for Layer 2 and Layer 3.
OpenFlow enabled ports	Lists the ports on the device that are enabled for OpenFlow.
Default action	Indicates the default action for packets that do not match any configured flows. By default, such packets are dropped. However, you can configure these packets to be sent to the controller by using the <b>default-behavior send-to-controller</b> command.
Maximum number of flows allowed	Indicates the maximum number of flows allowed on the device that is configured by using the <b>system-max openflow-flow-entries</b> command.

## History

Release version	Command history
6.0.1a	This command was modified to include the rbridge-id keyword.

# show openflow controller

Displays the status of the OpenFlow controller.

## Syntax

```
show openflow controller [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the status of the OpenFlow controller.

```
device# show openflow controller
Controller      Mode      TCP/SSL  IP-address  Port      Status      Role
Passive        PASSIVE  TCP      127.0.0.1   6633     TCP_LISTENING  EQUAL
BVC            ACTIVE   TCP      10.24.82.10 6633     OPENF_ESTABLISHED  EQUAL
BVC-BKUP      ACTIVE   SSL      10.24.82.11 6633     INIT          EQUAL
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show openflow flow

Displays the OpenFlow flow that are configured on the devices with IPv4 fields at the global level.

## Syntax

```
show openflow flow [ flow-id flow-id | interface bridge-id/slot/port ]
```

## Parameters

*flow-id*

Displays for a specific Flow ID.

**interface**

Displays OpenFlow flow entries for an interface.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode



## Examples

The **show openflow flow** command has the match capability as below.

Layer 2: Port, Source MAC, Destination MAC, Ether type, VLAN, VLAN PCP

Layer 3 : Port, VLAN, VLAN PCP, Ether type (IP, ARP and LLDP), Source IP, Destination IP, IP Protocol, IP ToS , IP Source Port, IP Destination Port, ICMPv4 Type, ICMPv4 code

Layer 2 flows:

```
device # show openflow flow
Logical Instance:                1
Datapath-ID:                    10027f8cad766
Total Number of data packets sent to controller: 10
Total Number of data bytes sent to controller : 1560
```

```
Total Number of Flows:          2
  Total Number of Port based Flows: 1
  Total Number of L2 Generic Flows: 1
  Total Number of L3 Generic Flows: 0
```

```
Total Number of Hardware entries for flows:      2
  Total Number of Hardware entries for Port flow: 1
  Total Number of Hardware entries for Generic flow: 1
```

```
Total Number of Openflow interfaces:    3
  Total Number of L2 interfaces:         3
  Total Number of L3 interfaces:         0
```

```
Flow ID: 1 Priority: 100 Status: ACTIVE
Rule:
  Ether type:                0x88cc
Instructions: Apply-Actions
  Action: FORWARD
    Out Port: OFPP_CONTROLLER
```

```
Statistics:
  Total Pkts: 10
  Total Bytes: NA
```

```
Flow ID: 2 Priority: 1900 Status: ACTIVE
Rule:
  In Port:                    Te 1/0/10
  In Vlan:                    Tagged[1789]
  Vlan Priority:               4
  Source Mac:                 0022.0033.abcd
  Destination Mac:           0123.cdef.ab44
  Ether type:                 0x800
Instructions: Apply-Actions
  Action: FORWARD
    Out Port: Te 1/0/24
    Vlan Priority: 5
```

```
Statistics:
  Total Pkts: 0
  Total Bytes: NA
```

Layer 3 flows:

```
device # show openflow flow
Logical Instance:                1
Datapath-ID:                    1000533e6ba00
Total Number of data packets sent to controller: 2508
Total Number of data bytes sent to controller : 414480
```

```
Total Number of Flows:          2
  Total Number of Port based Flows: 1
```

show openflow flow

```

Total Number of L2 Generic Flows: 1
Total Number of L3 Generic Flows: 0

Total Number of Hardware entries for flows: 2
Total Number of Hardware entries for Port flow: 1
Total Number of Hardware entries for Generic flow: 1

Total Number of Openflow interfaces: 5
Total Number of L2 interfaces: 0
Total Number of L3 interfaces: 5

Flow ID: 1 Priority: 100 Status: ACTIVE
Rule:
  Ether type: 0x88cc
  Instructions: Apply-Actions
  Action: FORWARD
        Out Port: OFPP_CONTROLLER

Statistics:
  Total Pkts: 10
  Total Bytes: NA

```

```

Flow ID: 2 Priority: 1900 Status: ACTIVE
Rule:
  In Port: Te 1/0/7
  Ether type: 0x800
  IP Protocol: 6
  IP Protocol Source Port: 13101
  IP Protocol Destination Port: 14101
  Source IPv4: 101.10.10.131
  Source IPv4 Mask: 255.255.255.255
  Destination IPv4: 201.10.10.131
  Destination IPv4 Mask: 255.255.255.255
  Instructions: Apply-Actions
  Action: FORWARD
        Out Group: 1
        Out Port: None
        Vlan Priority: 4, TOS : 44

Statistics:
  Total Pkts: 0
  Total Bytes: NA

```

**TABLE 19** Output fields for the show openflow flow command

Field	Description
Logical Instance	Indicates the logical instance of OpenFlow controller on the device.
Data path ID	Displays the data path ID assigned to the device.
Total Number of Flows	The total number of flows on the device.
Total number of data packets sent to controller	The number of packets sent to the controller.
Total number of data bytes sent to controller	The number of bytes sent to the controller.
Flow ID	An identifier for each flow. You can use the flow ID from this output to display flow-specific details.
Priority	The priority of the flow set by the controller when the flow is added, in the range 0 to 65536. If the priority value was not specified, the Extreme device will assign the default value, 32768.
Status	Indicates whether the flow is configured correctly in the device. A correctly configured flow will have its status as active.
Rule	Specifies the matching rule for the flow.
Instruction	Applies the specified actions immediately.
Statistics	Indicates the counter of packets and bytes.

# show openflow group

Displays all the groups in a flow for an OpenFlow port.

## Syntax

```
show openflow group [group-id]
```

## Parameters

*group-id*

Shows details of a specific OpenFlow group.

## Modes

Privileged EXEC mode

## Usage Guidelines

The hardware resources are shared between OpenFlow and other features, so these resources are allocated on a first-come-first-serve basis.

## Examples

The command displays the minimum and maximum traffic rates for each ports.

```
device# show openflow group
Max number of total groups           : 256
Max number of buckets per group      : 64
Max number of actions per bucket     : 3

Max number of groups (ALL/SELECT/INDIRECT) : 256 / 256 / 256
Max number of buckets/group (ALL/SELECT/INDIRECT) : 64 / 8 / 64

TOTAL number of groups (Type:ALL) in the system : 0
TOTAL number of groups (Type:SELECT) in the system : 0
TOTAL number of groups (Type:Indirect) in the system : 1

TOTAL number of groups in the system : 1

Group id 2
Transaction id      5694
Type                2
Packet Count       0
Byte Count         0
Flow Count         0
Number of buckets  1
bucket # 1
Weight              1
Number of actions  3
  action 0: out port: 786
  action 1: out port: 0011.2233.4455
  action 2: out port: Te 3/0/27
```

**TABLE 20** Output fields for the show openflow group command

Field	Description
Total number of groups	Total number of group of all types available on the flow, e.g. All, Indirect and Select.
Group ID	Displays the group ID number.
Transaction ID	Unique transaction ID for the specified group ID.
Type	Group type.
Packet count	The number of packets sent in the group.
Byte count	The number of bytes in the group.
Flow count	The number of flow in the group.
Number of buckets	Number of buckets per group.
Number of actions	Number of actions per bucket.

# show openflow interface

Displays the detailed interface configuration and capabilities of all interfaces or for a specific interface associated to a OpenFlow logical instance.

## Syntax

```
show openflow interface
```

## Parameters

### Tengigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds.

## Modes

Privileged EXEC mode

RBridge ID configuration mode

## Usage Guidelines

LLDP needs to be explicitly disabled to display OpenFlow enabled interfaces.

## Examples

To disable the LLDP, enter the following command.

```
device(config)#interface TenGigabitEthernet 12/0/12
device(conf-if-te-12/0/12)# lldp disable
device(conf-if-te-12/0/12)# openflow logical-instance 1
```

To show OpenFlow interface, enter the following command:

```
device # show openflow interface
-----
Port          Log-Ins   Link   Port-State  Speed   MAC           OF-portid  Mode
-----
Te 12/0/1     1         Up     Link down   10G    0005.33e5.c946  1          L2
Te 12/0/2     1         Up     Link down   10G    0005.33e5.c947  2          L2
Te 12/0/3     1         Up     Link down   10G    0005.33e5.c948  3          L2
```

**TABLE 21** Output fields of the show openflow interface command

Field	Description
Port	Indicates the port number on the device.
Log-Ins	Indicates the log-in numbers.
Link	Indicates the link status.
Port-State	Indicates the action to be performed on packets that reach the interface.
Speed	Indicates the port speed.
MAC	Indicates the MAC address of the port.

**TABLE 21** Output fields of the show openflow interface command (continued)

Field	Description
OF-PortID	Indicates the OpenFlow port ID that is assigned to the port on the device. Port numbers on the device are mapped to OpenFlow port IDs.
Mode	Indicates the OpenFlow mode enabled on the port.

# show openflow meter

Displays all the meters in a flow for an OpenFlow port.

## Syntax

```
show openflow meter [meter-id]
```

## Parameters

*meter-id*

Shows details of a specific OpenFlow meter.

## Modes

Privileged EXEC mode

## Usage Guidelines

The hardware resources are shared between OpenFlow and other features, so these resources are allocated on a first-come,-first-served basis.

## Examples

```
Device# show openflow meter
TOTAL Meters: 2
Meter id: 1
  Transaction id:      6245
  Meter Flags:        KBPS
  Flow Count:         0
  Number of bands:    1
  In packet count:    NA
  In byte count:      0
  Band Type:         DROP
    Rate:              150000
    Burst size:        1
    Prec level:        0
    In packet band count: NA
    In byte band count: 0
Meter id: 2
  Transaction id:      6350
  Meter Flags:        KBPS
  Flow Count:         0
  Number of bands:    1
  In packet count:    NA
  In byte count:      0
  Band Type:         DROP
    Rate:              370000
    Burst size:        4
    Prec level:        0
    In packet band count: NA
    In byte band count: 0
device#
```

**TABLE 22** Output fields for the show openflow meter command

Field	Description
Total number of meters	Total number of meters available in the flow.
Meter ID	Displays the meter ID number.
Transaction ID	Unique transaction ID for the specified meter ID.
Meter flags	Metering capability.
Flow count	The number of flow in the meter.
Number of bands	Number of bands per meter.
Band type	Band type supported on the meter. Supported band type is DROP.



# show openflow queues

Displays the queue entries for an interface for an OpenFlow port.

## Syntax

```
show openflow queues [ interface rbridge-id/slot/port ]
```

## Parameters

*interface*

Displays OpenFlow queue entries for an interface.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Ensure that OpenFlow queueing is configured on the device. You can associate a flow with an **OFFPAT\_ENQUEUE** action which forwards the packet through the specific queue on a port.

## Examples

The command displays the minimum and maximum traffic rates for each ports.

```
device# show openflow queues
Openflow Port      Te 0/1
Queue 0
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 0
  Tx Bytes: : 0
Openflow Port      Te 0/1
Queue 1
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 0
  Tx Bytes: : 0
Openflow Port      Te 0/1
Queue 2
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 0
  Tx Bytes: : 0
Openflow Port      Te 0/1
Queue 3
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 0
  Tx Bytes: : 0
Openflow Port      Te 0/1
Queue 4
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 0
  Tx Bytes: : 0
Openflow Port      Te 0/1
Queue 5
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 0
  Tx Bytes: : 0
Openflow Port      Te 0/1
Queue 6
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 0
  Tx Bytes: : 0
Openflow Port      Te 0/1
Queue 7
  Min Rate: 0 bps          Max Rate: 0 bps
  Tx Packets: 37570
  Tx Bytes: : 3569150
```

**TABLE 23** Output fields for the show openflow queues command

Field	Description
OpenFlow port	Slot and port number.
Queue	Displays the queue number for the specified OpenFlow port.
Rate	Minimum and maximum rate of the flow for the queue.
Packet	The number of transmitted packets in the queue.
Byte	The number of transmitted bytes in the queue.

# show openflow resources

Displays the OpenFlow usage of the resources at the global level.

## Syntax

**show openflow resources**

## Modes

Privileged EXEC mode

## Examples

```
device# show openflow resources
Used - Number of HW entries consumed
Free - Number of Port based flows that can be successfully programmed

Slot: 1          Module: LC48x10G
Openflow L2 Flows: MAX: 4096      Used: 0          Free: 4096
Openflow L3 Flows: MAX: 4096      Used: 96         Free: 4000

Slot: 3          Module: LC12X40G
Openflow L2 Flows: MAX: 4096      Used: 0          Free: 4000
Openflow L3 Flows: MAX: 4096      Used: 12         Free: 4084

Openflow Meter Resources:          MAX: 768          Used: 24          Free: 744
Openflow Group Resources:
      ALL          MAX: 256          Used: 8           Free: 248
      INDIRECT    MAX: 256          Used: 8           Free: 248
      SELECT      MAX: 256          Used: 8           Free: 248
```

**TABLE 24** Output fields for the show openflow resources command

Field	Description
Used	Number of hardware entries.
Free	Number of port based flows, that can be programmed.
Slot	Indicates the slot number
Module	Indicates the device number
OpenFlow flows	Available, used and maximum number of OpenFlow flows for Layer 2 and Layer 3.
OpenFlow Meter resources	Available, used and maximum number of OpenFlow meters for the device.
OpenFlow Group resources	Available, used and maximum number of OpenFlow group (All, Indirect and Select) for the device.

# show overlapping-vlan-resource usage

Shows the utilization of the hardware table entries that support classified or transport VLAN classifications that use overlapping CTAGs in a Virtual Fabrics context.

## Syntax

```
show overlapping-vlan-resource usage
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is platform-specific. For platforms that do not have such a table for classified or transport VLANs, the percentage is zero.

## Examples

The following example illustrates overlapping VLAN resource usage for Brocade VDX 6740 series platforms:

```
device# show overlapping-vlan-resource usage
Number of table entries used:50.04%(max 4003, used 2003)
```

The following example illustrates overlapping VLAN resource usage for Brocade VDX 6940 series platforms:

```
device# show overlapping-vlan-resource usage
Number of table entries used:84.44%(max 1774, used 1498)
Number of table entries used:56.31%(max 1774, used 999)
Number of table entries used:84.44%(max 1774, used 1498)
Number of table entries used:0.00%(max 1774, used 0)
Number of table entries used for EXM 0 :39.55%(max 15564, used 6156)
Number of table entries used for EXM 1 :6.43%(max 15564, used 1000)
Number of table entries used for EXM 2 :13.18%(max 15564, used 2052)
Number of table entries used for EXM 3 :0.00%(max 15564, used 0)
```

# show overlay-gateway

Displays status and statistics for the VXLAN overlay-gateway instance.

## Syntax

```
show overlay-gateway [ name name [ vlan statistics ] ] [ rbridge-id rbridge-id ] [ statistics ]
```

## Parameters

*name*

Name of the configured VXLAN gateway. Network OS supports only one gateway instance.

**vlan statistics**

Displays statistics for each VLAN for the VXLAN gateway. Statistics include transmitted and received packet counts and byte counts exchanged for each exported VLAN. Because each exported VLAN maps to a VXLAN, these statistics apply on a per-VXLAN-counters basis. Per-VLAN counters are not enabled by default. You need to first run the **enable statistics direction** command for the gateway to enable statistics for specified VLAN IDs.

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**statistics**

Displays statistics for the VXLAN gateway. Statistics include transmitted and received packet counts and byte counts. These counters are derived by aggregating tunnel counters for all the tunnels of the gateway.

## Modes

Privileged EXEC mode

## Usage Guidelines

Output includes the gateway name, the system-assigned gateway ID, source IP address, VRF, administration state, number of tunnels associated, and the Rbridge IDs on which the gateway is configured.

If you specify the gateway name, the gateway must already be configured.

## Examples

To show the status for a gateway instance that is configured for an NSX Controller:

```
device# show overlay-gateway

Overlay Gateway "gateway1", ID 1, rbridge-ids 22-23
Type nsx, Admin state up, Tunnel mode VXLAN
IP address 10.10.10.1 ( ve1000, Vrid 100 ), Vrf default-vrf
Number of tunnels 2
Packet count: RX 0           TX 0
Byte count   : RX (NA)      TX 0
```

show overlay-gateway

To show the status for a gateway instance that is configured for Layer 2 extension with a loopback interface:

```
device# show overlay-gateway

Overlay Gateway "gateway1", ID 1, rbridge-ids 22-23
Type layer2-extension, Admin state up
IP address 10.10.10.1 (Loopback 10), Vrf default-vrf
Number of tunnels 2
Packet count: RX 0           TX 0
Byte count  : RX (NA)       TX 0
```

To show statistics for the gateway instance:

```
device# show overlay-gateway statistics

Gateway Name      RX packets      TX packets      RX bytes      TX bytes
=====
GW1                200000          10000           22227772     1110111
```

To display statistics for VLANs attached to the VXLAN gateway:

```
device# show overlay-gateway name GW1 vlan statistics

VLAN  VNI    Tx    Rx    Packets    Bytes
-----
      Tx    Rx    Tx    Rx
-----
10     1010  10000 200000  1110111  22227772
11     1011   2200    -    221334    -
21     1021    -     1     -        -    100
```

```
sw0# show overlay-gateway name test vlan statistics

VLAN ID  RX packets    TX packets
=====
30        0              0
40       3696         3696
```

# show ovldb-server

Displays details of Open vSwitch Database (OVSLB) SSL server configurations.

## Syntax

```
show ovldb-server [ name ]
```

## Parameters

*name*

Specifies an OVSLB SSL server.

## Modes

Privileged EXEC mode

## Usage Guidelines

### NOTE

Currently only one server can be configured.

## Examples

To view details for all configured servers:

### NOTE

Currently only one server can be configured.

```
device# show ovldb-server
name: myserver
port: 6640
Connection 1:
Remote IP: 10.10.10.1
Up-time: 12:00:00
RPCs: 1000000
ECHOs: 1000000
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show policymap

Displays configured policy-maps and class-map Policer parameters applied to switch interfaces.

## Syntax

```
show policymap [ interface <N>gigabitethernet rbridge-id/slot/port input | output ] [ details policyname ]
```

## Parameters

**interface** **tengigabitethernet** *rbridge-id/slot/port*  
Interface where policy-map is bound.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace **<N>gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**input | output**

Direction (inbound or outbound) where the policy-map is applied.

**details** *policyname*

Displays the detail configuration of the policy-map along with binding information.

## Modes

Global configuration mode

Interface subtype configuration mode

## Usage Guidelines

Enter **show policymap** for a specific interface to display the policy-map binding settings (policy-map name and traffic direction), police-priority-map applied, and class-map Policer parameters applied for that interface.

Enter **show policymap** without identifying an interface and direction of traffic to display policy-map binding for all interfaces on the switch.

### NOTE

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

The following are definitions of terms used in output from the **show policymap** command:

- Interface: The interface for which rate limiting information is being displayed.



- Direction: The traffic direction for which rate limiting is applied.
- police-priority-map: Remarked priority-map used for Policer application (802.1 p priority remarked map).
- Conform: The traffic in bytes that has been forwarded from this interface that is within the CIR bandwidth limits.
- Exceeded: The traffic that has been exceeded the bandwidth available in the CIR limits and has not exceed the EIR limits for this rate-limit policy.
- Violated: The traffic that has exceeded the bandwidth available in the CIR and EIR limits.
- set-dscp: The DSCP value which is applied to the traffic for the given color (conform, exceed, violate).
- set-tc: The remapped traffic class queue for the traffic for the given color (conform, exceed, violate).
- Total: The total traffic in bytes carried on this interface for the defined rate-limit policy.

## Examples

To display policy-map binding and class-map parameters applied to a specific interface:

```
switch# show policymap interface tengigabitethernet 4/1 input
Interface : Ten Gigabit Ethernet 4/1
Policymap: policymapA-1
Direction: Input
Input Excluded lossless priorities: None

Class-map: default
  Police:
    cir 5 bps cbs 5678 bytes eir 512000 bps ebs 4096 bytes
    Police-priority-map: po-pr-map1
    Conformed: 30720 bytes set-dscp 0 set-tc 0
    Exceeded: 23424 bytes set-dscp 0 set-tc 0
    Violated: 0 bytes
    Total: 54144 bytes
```

To display policy-map binding information for all switch interfaces:

```
switch# show policymap
Interface : Ten Gigabit Ethernet 4/2
Inbound policy map is policymapA-1
Outbound policy map is not set
Interface : Ten Gigabit Ethernet 4/3
Inbound policy map is not set
Outbound policy map is not set
Interface : Ten Gigabit Ethernet 4/4
Inbound policy map is not set
Outbound policy map is not set
```

show port port-channel

# show port port-channel

Displays the detailed LACP attributes that are configured and negotiated with its partner.

## Syntax

```
show port port-channel port_id
```

## Parameters

*port\_id*

Port to display. The number of available channels range from 1 through 6144.

## Modes

Privileged EXEC mode

# show port-channel

Displays the Link Aggregation Group (LAG) information for a port-channel.

## Syntax

```
show port-channel [ channel-group-number | detail | load-balance | summary ]
```

## Parameters

*channel-group-number*

Specifies a LAG port channel-group number to display. The number of available channels range from 1 through 6144.

**detail**

Displays detailed LAG information for a port-channel, including PXE information if available.

**load-balance**

Displays the load-balance or frame-distribution scheme among ports in the port-channel.

**summary**

Displays the summary information per channel-group.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the LAGs present on the system with details about the LACP counters on their member links. LAG interfaces are called port-channels.

If you do not specify a port-channel, all port-channels are displayed.

When using the **show port-channel** *channel-group-number* command, an asterisk in the command output designates that the designated port channel is the primary link through which the BUM (Broadcast, Unknown unicast and Multicast) traffic flows.

show port-channel

## Examples

The following example displays a detailed port-channel information.

```
device# show port-channel detail

LACP Aggregator: Po 10 (vLAG) (Defaulted)
Aggregator type: Standard
Ignore-split is enabled
Member rbridges:
rbridge-id: 1 (2)
rbridge-id: 2 (1)
Actor System ID - 0x8000,01-e0-52-00-00-01
Admin Key: 0010 - Oper Key 0010
Receive link count: 1 - Transmit link count: 1
Individual: 0 - Ready: 1
Partner System ID - 0x8000,00-05-1e-cd-0f-ea
Partner Oper Key 0010
Member ports on rbridge-id 2:
Link: Te 2/0/7 (0x218038006) sync: 1
```

# show port-channel-redundancy-group

Displays the port-channel redundancy-group information.

## Syntax

```
show port-channel-redundancy-group group-id
```

## Parameters

*group-id*

Designates which port-channel to display.

## Modes

Privileged EXEC mode

## Examples

Typical command execution example:

```
switch#show port-channel-redundancy-group 1
Group ID                : 1
Member Ports            : Port-channel 5, Port-channel 7
Configured Active Port-channel: Port-channel 5
Current Active Port-channel  : Port-channel 5
Backup Port-channel     : Port-channel 7
```

Typical command output, when the backup vLAG is operationally down or not yet created.

```
switch#show port-channel-redundancy-group 1
Group ID : 1
Member Ports            : Port-channel 5, Port-channel 7
Configured Active Port-channel: Port-channel 5
Current Active Port-channel  : Port-channel 5
Backup Port-channel     : None
```

Typical command output, when both vLAG members are operationally down.

```
switch#show port-channel-redundancy-group 1
Group ID                : 1
Member Ports            : Port-channel 5, Port-channel 7
Configured Active Port-channel: Port-channel 5
Current Active Port-channel  : None
Backup Port-channel     : None
```

Typical command output, when the active vLAG is not configured.

```
switch#show port-channel-redundancy-group 1
Group ID                : 1
Member Ports            : Port-channel 5, Port-channel 7
Configured Active Port-channel: None
Current Active Port-channel  : Port-channel 5
Backup Port-channel     : Port-channel 7
```

# show port-profile

Displays the AMPP port-profile configuration information.

## Syntax

`show port-profile`

## Modes

Privileged EXEC mode

## Examples

Example of this command:

```
switch# show port-profile

port-profile default
ppid 0
  vlan-profile
  switchport
  switchport mode trunk
  switchport trunk allowed vlan all
port-profile auto-dvPortGroup-2
ppid 1
  vlan-profile
  switchport
  switchport mode trunk
  switchport trunk allowed vlan add 45
port-profile auto-dvPortGroup-1
ppid 2
  vlan-profile
  switchport
  switchport mode trunk
  switchport trunk allowed vlan add 3-10
```

## Related Commands

[show running-config](#), [show port-profile interface](#)

# show port-profile domain

Displays the status of Automatic Migration of Port Profiles (AMPP) profiles and port-profile domains.

## Syntax

```
show port-profile [ port-profile-name ] | domain port-profile-domain-name ] [ status | activated | applied ] | associated ]
```

## Parameters

*port-profile-name*

The name of a port-profile.

**domain**

Enables specification of a port-profile domain name.

*port-profile-domain-name*

Name of a port-profile domain.

**status**

Enables selection of status type.

**activated**

Specifies all port-profiles with the activated status.

**applied**

Specifies all port-profiles with the applied status.

**associated**

Specifies all port-profiles with the associated status.

## Modes

Privileged EXEC mode

## Usage Guidelines

Enter **show port-profile status** to display the status of all AMPP profiles.

If **no** option is specified, then all port-profiles that match the criteria are shown.

## Examples

The following example shows the status of all port-profiles:

```
switch# show port-profile status
Port-Profile      PPID  Activated  Associated MAC  Interface
auto-dvPortGroup-2  1     Yes       0050.5681.2ed5  none
                  0050.5699.5524  te0/2
                  0050.5699.39e0  te0/1
auto-dvPortGroup-1  2     Yes       0050.5681.083c  none
```

show port-profile domain

The following example shows the status of a port-profile domain:

```
switch# show port-profile domain vDC1_Domain status
Port-Profile  PPID  Activated  Associated MAC  Interface
Tenant1_PP    1      No        None           None
Tenant2_PP    2      No        None           None
```



# show port-profile interface

Displays AMPP port-profile information for interfaces.

## Syntax

```
show port-profile interface [ all | port-channel channel-group-number | <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### all

Displays the port-profile information for all interfaces.

### port-channel *channel-group-number*

Specifies a LAG port channel-group number to display. Valid values range from 1 through 6144.

### <N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display AMPP port-profile information for either all interfaces, or for specific interfaces.

## Related Commands

[show running-config](#), [show port-profile](#)

# show port-profile name

Displays the port profile information for a named port-profile.

## Syntax

```
show port-profile name port-profile-name { qos | security | status | vlan }
```

```
show port-profile name port-profile-name name port-profile-name validate
```

## Parameters

*port-profile-name*

The name of the port-profile. The maximum number of characters is 64.

**qos**

QoS sub-profile

**security**

Security sub-profile

**status**

Specific port-profile status

**vlan**

VLAN sub-profile

**validate**

Validates two port-profiles against each other.

## Modes

Privileged EXEC mode

## Examples

Typical command output for the **show port-profile name** command:

```
device# show port-profile name default
port-profile default
  ppid 0
  vlan-profile
    switchport
    switchport mode trunk
    switchport trunk native-vlan 1
```

# show port-security

Displays the configuration information related to port security.

## Syntax

```
show port-security
```

## Modes

Privileged EXEC mode

## Examples

```
switch# show port-security
Secure  MaxSecureAddr      CurrentAddr      StaticSec  Violated   Action      OUIs      Sticky
Port    (count)                (count)         (count)
Te 1/1   2                      1                3          No        Restrict   2         No
Te 1/3   3                      3                5          Yes       Shutdown  0         Yes
```

# show port-security addresses

Displays the configuration information related to port-security addresses.

## Syntax

`show port-security addresses`

## Modes

Privileged EXEC mode

## Examples

```
switch# show port-security addresses
                Secure Mac Address Table
-----
Vlan           Mac Address      Type              Ports
1              0000.0000.0001   Secure-Dynamic    1/1
1              0000.0000.0002   Secure-Static     1/2
```

# show port-security interface

Displays the configuration information related to port-security interfaces.

## Syntax

```
show port-security interface [ all | port-channel channel-group-number | <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### all

Displays the port-security information for all interfaces.

### port-channel *channel-group-number*

Specifies a LAG port channel-group number to display. Valid values range from 1 through 6144.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show port-security interface TenGigabitEthernet 1/1
Port Security           :Enabled
Port Status             :up / Down (Security Violated)
Violation Mode         :Restrict
Violated                :Yes/No
Sticky enabled         :Yes/No
Maximum MAC Addresses  :2
Total MAC Addresses    :2
Configured MAC Addresses :5
Violation time         : Fri Mar 22 05:53:03 UTC 2013
Shutdown time(in Minutes) :5
Number of OUIs configured :1
```

show port-security oui interface

# show port-security oui interface

Displays the configuration information related to port security for Organizationally Unique Identifier (OUI) interfaces.

## Syntax

`show port-security oui interface`

## Modes

Privileged EXEC mode

## Examples

```
switch# show port-security oui interface TenGigabitEthernet 1/1
OUIs configured      : 3
OUIs                 : 0010.0a00.0000
                   : 0020.0b00.0000
                   : 0030.0c00.0000
```

# show port-security sticky interface

Displays the configuration information related to port security for a sticky interface.

## Syntax

```
show port-security sticky interface
```

## Modes

Privileged EXEC mode

## Examples

```
switch# show port-security sticky interface TenGigabitEthernet 1/1
VlanId  Mac-address      Type      State      Ports
1         0000.0000.1111      Secure-Sticky  Active      Te 1/1
```

# show process cpu

Displays information about the active processes in the switch and their corresponding CPU utilization statistics.

## Syntax

```
show process cpu [ rbridge-id { rbridge-id | all } ] [ summary ] [ history ] [ top ] [ all-partitions ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridges.

### summary

Displays a summary view of cpu usage.

### history

Displays the history of CPU usage.

### top

Displays current CPU utilization.

### all-partitions

Displays a summary view of all partitions.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

For an explanation of process states, refer to the UNIX manual page for the **ps** command.

## Examples

To show the information for all processes:

```
switch# show process cpu summary
```

```
Realtime Statistics:
```

```
Total CPU Utilization: 0% (user procs:0%, system-kernel:0%, iowait:0%)
```

```
Load Average: One minute: 0.00; Five minutes: 0.03; Fifteen minutes: 0.01
```



To show CPU usage information by individual processes:

```
switch# show process cpu
```

```
Realtime Statistics:
Total CPU Utilization: 0% (user procs:0%, system-kernel:0%, iowait:0%)
Load Average: One minute: 0.00; Five minutes: 0.02; Fifteen minutes: 0.00
Active Processes Lifetime Statistic:
  PID   Process          CPU%  State   Started
17169  sh                 1.00  S       13:44:27 Jul  1, 2012
 2060  emd                 0.80  S       21:52:27 Jun 29, 2012
 2462  SWITCH_TMR_0       0.60  S       21:53:08 Jun 29, 2012
17170  imishow_proc_cp    0.50  S       13:44:27 Jul  1, 2012
 2207  ospfd               0.20  S       21:52:41 Jun 29, 2012
 2211  mstpd               0.20  S       21:52:41 Jun 29, 2012
 2208  rtmd                0.10  S       21:52:41 Jun 29, 2012
(Output truncated)
```

To show the information for all partitions:

```
switch# show process cpu all-partitions
Load Average:
Blade   1-min   5-min   15-min
M1:    2.27   2.21   2.08
M2:    2.02   2.07   2.02
L1/0:  2.81   2.27   2.15
L1/1:  2.00   2.00   2.00
L2/0:  2.00   2.01   2.00
L2/1:  2.06   2.03   2.00

Total CPU Utilization (in %):
Blade   current   user   system   iowait
M1:    1.05    0.28   0.47    0.29
M2:    0.72    0.18   0.25    0.29
L1/0:  4.39    0.14   3.83    0.41
L1/1:  0.5     0.00   0.08    0.42
L2/0:  0.49    0.01   0.05    0.44
L2/1:  0.5     0.01   0.05    0.44
```

To show the information for all partitions for a Top of Rack device:

```
switch# show process cpu all-partitions
Load Average:
Blade   1-min   5-min   15-min
SW/0:   2.45   2.23   2.18
SW/1:   2.00   2.00   2.00

Total CPU Utilization (in %):
Blade   current   user   system   iowait
SW/0:   18.02    0.95   16.67    0.40
SW/1:   1.5     0.54   0.48    0.48
```

## Related Commands

[show process memory](#), [show process info](#)

# show process info

Displays system processes hierarchically.

## Syntax

```
show process info [ rbridge-id { rbridge-id | all } ]
```

## Command Default

This command is executed on the local switch.

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Pagination is not supported with this command. Use **more** in the terminal window to display the output one page at a time.

This command is supported only on the local switch.

## Examples

To display system processes hierarchically:

```
switch# show process info
```

```
PID      CMD
2        kthreadd
3        \_ migration/0
4        \_ ksoftirqd/0
5        \_ watchdog/0
6        \_ migration/1
7        \_ ksoftirqd/1
8        \_ watchdog/1
9        \_ migration/2
10       \_ ksoftirqd/2
11       \_ watchdog/2
12       \_ migration/3
13       \_ ksoftirqd/3
14       \_ watchdog/3
15       \_ migration/4
16       \_ ksoftirqd/4
17       \_ watchdog/4
18       \_ migration/5
19       \_ ksoftirqd/5
20       \_ watchdog/5
21       \_ migration/6
22       \_ ksoftirqd/6
(Output truncated)
```

## Related Commands

[show process cpu](#), [show process memory](#)

# show process memory

Displays the memory usage information based on processes running in the system.

## Syntax

```
show process memory [ rbridge-id { rbridge-id | all } ] [ summary ]
```

## Parameters

### **rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

### **summary**

Displays a summary view of memory usage.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local device.

In the output of this command, "Low" and "High" refer to the Linux "LOWMEM" and "HIGHMEM" definitions of memory space. Generally speaking, low memory is always mapped into the kernel's address space, whereas high memory is used for user applications. The user is expected to be familiar with Linux memory allocations.

## Examples

To show memory usage information by individual processes:

```
device# show process memory

Rbridge-id: 54
%Memory Used: 48.9778%; TotalMemory: 3354812 KB; Total Used: 1643112 KB
Total Free: 1711700 KB; Low Free: 759700 KB; High Free: 500156 KB; Cached: 406696 KB
  PID Process      MEM%      VSIZE(KB)      RSS(KB)      PSS(KB)
  2895 postgres      4.30      203788      147224      101079
  2682 Dcmd.Linux.powe  4.30      324416      145956      119904
  3063 ospf6d          2.80      375424      94816       51118
  3042 rps            2.70      266576      93800       73916
  2890 postgres      2.60      155492      89616       46558
  3039 ospfd          2.50      367280      86068       42306
  3038 bgpd           2.40      372888      81520       37625
  3050 vrrpd          1.90      289480      65700       21929
  3040 ribmgr        1.90      203156      65004       45139
  4370 fibagt        1.60      184816      55772       36110
  3041 srm           1.60      183608      54272       34579
  3044 pimd          1.50      288724      53040       33215
  1264 confd        1.40      58856       46984       45873
  2336 raslogd       1.30      192276      44488       21016
  3055 iphelpd       1.30      232224      44360       24545
  3034 nsm           1.30      265352      43720       23292
  3054 arpd          1.20      209820      41276       21638
  3049 mstpd         1.10      200420      39768       19836
  3061 toamd         1.10      189932      38592       18752
  3064 tnlmgrd       1.10      181312      37580       18049
  2965 snmpd         1.00      135320      34012       11760
  3046 ssmd         0.90      179288      31504       11510
  4366 l2agtd        0.90      165984      31304       11608
  3045 mldd         0.80      184660      29948       9992
  3036 l2sysd        0.80      225680      29688       9596
  3047 qosd         0.80      175688      29068       9141
  3052 igmpd         0.80      176268      28684       8803
  3043 radv         0.80      164956      27352       7562
  4368 mcagtd       0.80      156732      27048       7351
  3056 onmd        0.70      175700      26812       6707
  3048 lacpd        0.70      166880      26280       6153
[output omitted, as will vary by device]

 3491 TD_TX_0          0.00      0           0           0
 4273 DCE_BLADE_THR_0  0.00      0           0           0
 4274 CBR_BH_43008080 0.00      0           0           0
 4275 CBR_HI_43008080 0.00      0           0           0
 4276 CBR_TX_43008080 0.00      0           0           0
 4277 CBR_RX_43008080 0.00      0           0           0
 4278 CBR_AS_43008080 0.00      0           0           0
 4279 CBR_MM_43008080 0.00      0           0           0
 4280 DCE_BLADE_CH_TH 0.00      0           0           0
 4282 FCOE_AGT_MI_THR 0.00      0           0           0
```

## Related Commands

[show process cpu](#), [show process info](#)

# show prom-access

Shows the Boot PROM access status.

## Syntax

```
show prom-access
```

## Command Default

The boot PROM is accessible.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to determine whether the Boot PROM is accessible.

Under non-FIPS compliant operation, you can access the Boot PROM by holding down the ESC key during the 4-second period when the switch is booting up. In FIPS compliant state, PROM access is disabled to prevent users from net-installing firmware.

If PROM access is enabled, you can disable it in preparation for FIPS compliance. If PROM access is disabled, you cannot re-enable it.

Enter the **unhide fips** command with password "fibranne" to make the command available.

## Examples

To view the Boot PROM access status:

```
switch# show prom-access  
  
PROM access Enabled
```

# show protected-ports

Displays the status of uplink ports and protected ports for their associated VLANs.

## Syntax

```
show protected-ports [ vlan VLAN-ID ]
```

## Parameters

**vlan** *VLAN-ID*  
Specifies a VLAN.

## Modes

Privileged EXEC mode

## Examples

The following example displays the status of uplink ports and protected ports.

```
device# show protected-ports
Vlan      Uplink Ports      Protected Ports
-----
10        TE1/0/1            TE1/0/2, TE1/0/3, TE1/0/4
6000     TE1/0/1, TE1/0/8  TE1/0/2, TE1/0/3, TE1/0/4, TE1/0/6
```

## History

Release version	Command history
7.2.0	This command was introduced.

# show qos flowcontrol interface

Displays all of the configured flow control information for an interface.

## Syntax

```
show qos flowcontrol interface [ <N>gigabitethernet rbridge-id/slot/port | all ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

### all

Reports QoS flow control statistics for all interfaces within the system.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the runtime state retrieved from the dataplane reflecting the operation of 802.3x pause or Priority Flow Control (PFC) on an interface.

The administrative state for pause generation and reception or processing is presented for the interface (802.3x mode) or for each CoS (PFC mode). TX\_Pause frame generation statistics are always presented for the interface. The RX\_Pause BitTimes is presented for the interface (802.3x mode) or for each CoS (PFC mode). When PFC is deployed under the CEE Provisioning model, then the command reports whether the Data Center Bridging eXchange protocol (DCBX) has overridden the user configuration, for example when the DCBX detects a mis-configuration between CEE peers, it disables PFC operationally.



## Examples

To display all of the configured flow control information for a 10-gigabit Ethernet interface:

```
switch# show qos flowcontrol interface tengigabitethernet 5/0/1
```

```
Interface Ten Gigabit Ethernet 5/0/1
Mode PFC
DCBX enabled for PFC negotiation
TX 0 frames
  TX      TX      RX      RX Output Paused
CoS Admin Oper Admin Oper 512 BitTimes
-----
  0  Off  Off  Off  Off          0
  1  Off  Off  Off  Off          0
  2   On  Off   On  Off          0
  3  Off  Off  Off  Off          0
  4  Off  Off  Off  Off          0
  5  Off  Off  Off  Off          0
  6  Off  Off  Off  Off          0
```

## Related Commands

[show qos interface](#), [show cee maps](#)

# show qos interface

Displays a summary of all QoS configurations applied on an interface.

## Syntax

```
show qos interface [ <N>gigabitethernet rbridge-id/slot/port | port-channel number | all ]
```

## Command Default

If no interface is specified, QoS information for all interfaces is displayed.

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### port-channel *number*

Specifies the port-channel of the interface. Valid values range from 1 through 63.

### all

Reports QoS configurations for all interfaces within the system.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display a summary of all QoS configuration applied on an interface, including QoS Provisioning mode, CEE map, Layer 2 priority, Traffic Class mapping, congestion control, and the scheduler policy.

## Examples

To display all of the configured QoS information for a 10-gigabit Ethernet interface:

```
switch# show qos interface tengigabitethernet 22/0/1

Interface Ten Gigabit Ethernet 22/0/1
  Provisioning mode cee
  CEE Map demo
  Default CoS 0
  Interface trust cos
  CoS-to-CoS Mutation map 'default'
-----
      In-CoS:  0  1  2  3  4  5  6  7
-----
  Out-CoS/TrafficClass: 0/4 1/4 2/6 3/4 4/4 5/4 6/4 7/4
  Tail Drop Threshold 1081344 bytes
  Per-CoS Tail Drop Threshold (bytes)
      CoS:      0      1      2      3      4      5      6      7
-----
  Threshold: 129761 129761 129761 129761 129761 129761 129761 129761
  Flow control mode PFC
  CoS2 TX on, RX on
  Multicast Packet Expansion Rate Limit 3000000 pkt/s, max burst 4096 pkts
  Multicast Packet Expansion Tail Drop Threshold (packets)
  TrafficClass:  0  1  2  3  4  5  6  7
-----
  Threshold:      64  64  64  64  64  64  64  64
  Traffic Class Scheduler configured for 0 Strict Priority queues
  TrafficClass:  0  1  2  3  4  5  6  7
-----
      DWRRWeight:  0  0  0  0  60  0  40  0
  Multicast Packet Expansion Traffic Class Scheduler
  TrafficClass:  0  1  2  3  4  5  6  7
-----
  DWRRWeight:      25  25  25  25  25  25  25  25
```

# show qos maps

Displays information on the defined QoS maps.

## Syntax

```
show qos maps [ cos-mutation [ name ] | cos-traffic-class [ name ] ]
```

## Command Default

Report shows all defined QoS maps.

## Parameters

**cos-mutation** *name*

Specifies to report on only the named CoS-to-CoS mutation QoS map.

**cos-traffic-class** *name*

Specifies to report on only the named CoS-to-Traffic Class QoS map.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display information on the QoS defined maps. For each QoS map, the configuration state is displayed with a list of all interfaces bound to the QoS map.

## Examples

To display information on the defined QoS maps:

```
switch# show qos maps

CoS-to-CoS Mutation map 'test'
  In-CoS:  0  1  2  3  4  5  6  7
-----
  Out-CoS:  0  1  2  3  5  4  6  7
Enabled on the following interfaces:
  Te 0/5
CoS-to-Traffic Class map 'test'
  Out-CoS:  0  1  2  3  4  5  6  7
-----
  TrafficClass:  0  1  2  3  5  4  6  7
Enabled on the following interfaces:
  Te 0/5
```

## Related Commands

[show qos interface](#)

# show qos maps dscp-cos

Displays configured DSCP to CoS mutation maps.

## Syntax

```
show qos maps dscp-cos
```

## Modes

Privileged EXEC mode

## Examples

To display information on defined QoS DSCP-CoS maps and application on interfaces.

```
device# show qos maps dscp-cos

Dscp-to-CoS map 'test' (dscp= d1d2)
d1 : d2 0 1 2 3 4 5 6 7 8 9
-----
0 :    00 03 07 03 07 03 07 03 07 01
1 :    01 05 06 05 06 05 06 05 06 02
2 :    02 02 02 02 03 03 03 03 03 03
3 :    03 03 04 04 04 04 04 04 04 04
4 :    05 05 05 05 05 05 05 05 06 06
5 :    06 06 06 06 06 06 07 07 07 07
6 :    07 07 07 07
Enabled on the following interfaces:
  Te 16/2/2
```

This information relates to the following map configuration applied to interface 16/2/2:

```
qos map dscp-mutation test
mark 1,3,5,7 to 3
mark 11,13,15,17 to 5
mark 12,14,16,18 to 6
mark 2,4,6,8 to 7
```

## Related Commands

[qos map dscp-cos](#), [show qos interface](#)

# show qos maps dscp-mutation

Displays configured DSCP mutation maps.

## Syntax

```
show qos maps dscp-mutation
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is only supported on VDX 8770-4, VDX 8770-8, and later switches.

## Examples

To display information on defined QoS DSCP mutation maps.

```
device# show qos maps dscp-mutation

Dscp-to-Dscp Mutation map 'test' (dscp= d1d2)
d1 :  d2 0  1  2  3  4  5  6  7  8  9
-----
0 :    00 09 10 09 10 09 10 09 10 09
1 :    10 19 20 19 20 19 20 19 20 19
2 :    20 21 22 23 24 25 26 27 28 29
3 :    30 31 32 33 34 35 36 37 38 39
4 :    40 41 42 43 44 45 46 47 48 49
5 :    50 51 52 53 54 55 56 57 58 59
6 :    60 61 62 63
Enabled on the following interfaces:
    Te 16/2/2
```

This information relates to the following map configuration applied to interface 16/2/2:

```
qos map dscp-mutation test
mark 1,3,5,7 to 9
mark 11,13,15,17 to 19
mark 12,14,16,18 to 20
mark 2,4,6,8 to 10
```

## Related Commands

[qos map dscp-mutation](#)

# show qos maps dscp-traffic-class

Displays configured DSCP to traffic class mutation maps.

## Syntax

```
show qos maps dscp-traffic-class
```

## Modes

Privileged EXEC mode

## Examples

To display information on defined QoS DSCP-Traffic-Class maps.

```
device# show qos maps dscp-traffic-class

Dscp-to-Dscp Mutation map 'test' (dscp= d1d2)
Dscp-to-Traffic Class map 'pqrs' (dscp= d1d2)
d1 :  d2 0  1  2  3  4  5  6  7  8  9
-----
0 :   00 03 07 03 07 03 07 03 07 01
1 :   01 05 06 05 06 05 06 05 06 02
2 :   02 02 02 02 03 03 03 03 03 03
3 :   03 03 04 04 04 04 04 04 04 04
4 :   05 05 05 05 05 05 05 05 06 06
5 :   06 06 06 06 06 06 07 07 07 07
6 :   07 07 07 07
Enabled on the following interfaces:
    Te 16/2/2
```

This information relates to the following map configuration applied to interface 16/2/2:

```
qos map dscp-mutation test
mark 1,3,5,7 to 3
mark 11,13,15,17 to 5
mark 12,14,16,18 to 6
mark 2,4,6,8 to 7
```

## Related Commands

[show qos interface](#)

# show qos queue interface

Displays the runtime state retrieved from the interface reflecting the number of packets and bytes sent and received for each priority.

## Syntax

```
show qos queue interface [ <N>gigabitethernet rbridge-id/slot/port | all ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N> gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### all

Reports QoS statistics for all Ethernet interfaces within the system.

## Modes

Privileged EXEC mode

## Examples

To display the queuing information for a 10-gigabit Ethernet interface:

```
switch# show qos queue interface tengigabitethernet 5/0/2
```

```
Interface Ten Gigabit Ethernet 5/0/2
```

CoS	RX		TC	TX	
	Packets	Bytes		Packets	Bytes
0	680458	87098624	0	0	0
1	0	0	1	32318	0
2	0	0	2	0	0
3	0	0	3	0	0
4	0	0	4	0	0
5	0	0	5	0	0
6	0	0	6	0	0
7	0	0	7	0	0

## Related Commands

[cee-map \(configuration\)](#), [qos map cos-mutation](#)  
[cee-map \(configuration\)](#), [qos map cos-mutation](#)



# show qos rcv-queue interface

Displays a summary of the runtime ingress queue state information applied to a Layer 2 interface.

## Syntax

```
show qos rcv-queue interface [ <N>gigabitethernet rbridge-id/slot/port | all ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### all

Reports QoS configurations for all 10-gigabit Ethernet interfaces within the system.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is not supported on Extreme VDX 8770-4 and VDX 8770-8 platforms.

## Examples

To display the runtime ingress queue state information retrieved from the dataplane for a 10-gigabit Ethernet interface:

```
device# show qos rcv-queue interface tengigabitethernet 22/0/2
```

```
Interface TenGigabitEthernet 22/0/2
In-use 404019 bytes, Max buffer 1081344 bytes
0 packets dropped
```

CoS	In-use Bytes	Max Bytes
0	0	1081344
1	0	1081344
2	404019	1081344
3	0	1081344
4	0	1081344
5	0	1081344
6	0	1081344
7	0	1081344

show qos rcv-queue interface

## Related Commands

[show qos rcv-queue multicast](#)

# show qos rcv-queue multicast

Displays the runtime state retrieved from the dataplane reflecting any multicast packet expansion packet drops resulting from a queue crossing the maximum queue depth.

## Syntax

```
show qos rcv-queue multicast [ <N>gigabitethernet rbridge-id/slot/port | all ]
```

## Parameters

<N> **gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**all**

Reports QoS multicast packet expansion receive queueing statistics for all ASICs within the system.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is not supported on VDX 8770-4 and VDX 8770-8 switches.

## Examples

To display the queueing information:

```
device# show qos rcv-queue multicast tengigabitethernet 1/0/2
```

```
Dropped Counts
Linecard/Portset          TC 0          TC 1          TC 2          TC 3
-----
0/0                        0            0            0            0
```

## Related Commands

[show qos rcv-queue interface](#)

# show qos red profiles

Displays configured Random Early Detect (RED) profiles.

## Syntax

**show qos red profiles**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

## Examples

Using **show qos red profiles** to display information on defined QoS RED profiles:

### NOTE

Notice that the first example shows output for the RED profile configured in the example for the **qos red profile** command.

```
device# show qos red profiles

Red Profile 2
  Minimum Threshold: 10
  Maximum Threshold: 80
  Drop Probability: 80
Activated on the following interfaces:
Te 1/2/2 Traffic-class: 7
Red Profile 100
  Minimum Threshold: 30
  Maximum Threshold: 80
  Drop Probability: 56
Activated on the following interfaces:
Te 1/1 Traffic-class: 2
Red Profile 200
  Minimum Threshold: 40
  Maximum Threshold: 60
  Drop Probability: 40
Activated on the following interfaces:
Te 1/1 Traffic-class: 4
```

Using **show qos interface** *interface-name* to examine the applied RED profiles for a specific interface:

```
device# show qos interface te 1/2/2

Interface Ten Gigabit Ethernet 1/2/2
  Provisioning mode non-cee
  Default CoS 0
  Interface COS trust untrusted
  CoS-to-CoS Mutation map 'default'
  CoS-to-Traffic Class map 'default'
      In-CoS:  0  1  2  3  4  5  6  7
-----
  Out-CoS/TrafficClass: 0/1 0/1 0/1 0/1 0/1 0/1 0/1 0/7
Interface DSCP trust untrusted
DSCP-to-DSCP Mutation map 'default' (dscp= d1d2)
d1 : d2 0  1  2  3  4  5  6  7  8  9
-----
  0 :   00 01 02 03 04 05 06 07 08 09
  1 :   10 11 12 13 14 15 16 17 18 19
  2 :   20 21 22 23 24 25 26 27 28 29
  3 :   30 31 32 33 34 35 36 37 38 39
  4 :   40 41 42 43 44 45 46 47 48 49
  5 :   50 51 52 53 54 55 56 57 58 59
  6 :   60 61 62 63
RED Enabled on the following Priorities:
  CoS: 7, Profile: 2
more
```

## Related Commands

[qos red profile](#)

# show qos red statistics interface

Displays Random Early Detect (RED) statistics for a specific interface.

## Syntax

```
show qos red statistics interface interface-name
```

## Parameters

*interface-name*

Name of interface where an RED profile is applied.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display statistics for RED on a specific interface where a RED profile is applied. Statistics include packets and bytes dropped for the CoS priority mapped to the profile for the interface.

## Examples

To display RED statistics on interfaces, use the **show qos red statistics interface** *interface-name* command. Notice that the colors in the following example (red, yellow, and green) relate to color-based priority mapping set through the Port-Based Policer feature. Refer to the *Network OS Administrator's Guide* for more information.

```
switch# show qos red statistics interface te 2/1

Statistics for interface: Te 2/1
Traffic-class: 2, ProfileId: 20
Packets Dropped: Red: 0, Yellow: 0, Green: 0, Queue Drops: 0
Bytes Dropped: Red: 0, Yellow: 0, Green: 0, Queue Drops: 0
Traffic-class: 3, ProfileId: 10
Packets Dropped: Red: 0, Yellow: 0, Green: 0, Queue Drops: 0
Bytes Dropped: Red: 0, Yellow: 0, Green: 0, Queue Drops: 0
```

## Related Commands

[qos red profile](#), [show qos red profiles](#)

# show qos tx-queue interface

Displays a summary of the runtime egress queue state information applied to a Layer 2 interface.

## Syntax

```
show qos tx-queue interface [ <N>gigabitethernet rbridge-id/slot/port | all ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

### all

Reports QoS configurations for all 10-gigabit Ethernet interfaces within the system.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **gigabitethernet** *rbridge-id/slot/port* parameter is used only on the VDX 8770-4, and VDX 8770-8 switches.

## Examples

To display the runtime egress queue state information retrieved from the dataplane for a tengigabit Ethernet interface:

```
device# show qos tx-queue interface tengigabitethernet 1/0/7
Interface TenGigabitEthernet 1/0/7 Transmit Queues
In-use 0 bytes, Max buffer 5046272 bytes
0 packets dropped
  TC      In-use      Max
  ----      Bytes      Bytes
  -----
  0         0         630784
  1         0         630784
  2         0         630784
  3         0         630784
  4         0         630784
  5         0         630784
  6         0         630784
  7         0         630784
```

show qos tx-queue interface

## History

Release version	Command history
5.0.0	This command was introduced.



# show radius servers

Displays the current status of the linked RADIUS servers.

## Syntax

**show radius servers**

## Modes

Privileged EXEC mode

## Command Output

The **show radius servers** command displays the following information:

Output field	Description
Server	The IP address of the RADIUS server.
Type	What type of functionality the RADIUS server provides.
Opens	The number of times the path to the RADIUS server opens.
Closes	The number of times the path to the RADIUS server closes.
Timeouts	The number of times the path to the RADIUS server times out.
Status	The current status of the path to the RADIUS server.

## Examples

Typical command output.

```
device# show radius servers
-----
Server                Type      Opens    Closes   Timeouts  Status
-----
10.37.73.21           any       100      0        0         dead
10.37.73.22           any       2787     2787     0         active
10.37.73.23           any       27       0        0         dead
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show rbridge-id

Displays the RBridge ID of each node that is configured in a Virtual Cluster Switching (VCS) cluster.

## Syntax

```
show rbridge-id [ swbd-number int | chassis { virtual-ip }
```

## Parameters

### swbd-number

Selects a switch type.

*int*

One or more integers (including a decimal) that identifies a switch type.

### chassis virtual-ip

Displays virtual IP addresses if configured

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to view the RBridge IDs of configured nodes in a VCS cluster, in addition to the switch type (SWBD) number and IPv4 and IPv6 virtual IP addresses.

## Examples

```
switch# show rbridge-id
RBRIDGE  SWBD
ID        NUMBER  V4   V6
-----
154      95         -   -
```

## Related Commands

[rbridge-id](#)

# show rbridge-running config

Displays the currently running configuration for an RBridge.

## Syntax

```
show rbridge-running-config rbridge-id rbridge-id
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies the RBridge ID whose configuration will be displayed.

## Modes

Privileged EXEC mode

## Examples

The following example shows partial output for this command:

```
switch# show rbridge-running-config rbridge-id 1
diag post rbridge-id 1
  enable
  !
dpod 1/0/1
  reserve
  !
dpod 1/0/2
  reserve
  !
dpod 1/0/3
  reserve
  !
dpod 1/0/4
  !
dpod 1/0/5
  !
dpod 1/0/6
  !
dpod 1/0/7
  !
[output truncated for brevity]

logging raslog console INFO
logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
logging syslog-facility local LOG_LOCAL7
switch-attributes 1
  chassis-name VDX6720-24
  host-name rb1
```

## Related Commands

[show global-running-config](#), [show rbridge-local-running-config](#)

# show rbridge-local-running-config

Displays the current local configuration for an RBridge.

## Syntax

```
show rbridge-local-running-config [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Examples

The following example shows partial output for this command:

```
switch# show rbridge-local-running-config rbridge-id 1
diag post rbridge-id 1
  enable
  !
dpod 1/0/1
  reserve
  !
dpod 1/0/2
  reserve
  !
dpod 1/0/3
  reserve
  !
dpod 1/0/4
  !
dpod 1/0/5
  !
dpod 1/0/6
  !
dpod 1/0/7
  !
dpod 1/0/8
  !
dpod 1/0/9
  !
dpod 1/0/10
  !
dpod 1/0/11
  !
dpod 1/0/12
  !
dpod 1/0/13
  !
dpod 1/0/14
  !
dpod 1/0/15
  !
dpod 1/0/16
  !
dpod 1/0/17
  !
dpod 1/0/18
  !
dpod 1/0/19
  !
dpod 1/0/20
  !
dpod 1/0/21
  !
dpod 1/0/22
  !
dpod 1/0/23
  !
dpod 1/0/24
  !

switch-attributes 1
  chassis-name VDX6720-24
  host-name rbl
  !
fabric route mcast rbridge-id 1
  !
rbridge-id 1
  ip route 0.0.0.0/0 10.17.0.1
  switch-attributes chassis-name VDX6720-24
  switch-attributes host-name rbl
  system-monitor fan threshold marginal-threshold 1 down-threshold 2
```

show rbridge-local-running-config

```
system-monitor fan alert state removed action raslog
system-monitor power threshold marginal-threshold 1 down-threshold 2
system-monitor power alert state removed action raslog
system-monitor temp threshold marginal-threshold 1 down-threshold 2
system-monitor cid-card threshold marginal-threshold 1 down-threshold 2
system-monitor cid-card alert state none action none
system-monitor sfp alert state none action none
system-monitor compact-flash threshold marginal-threshold 1 down-threshold 0
system-monitor MM threshold marginal-threshold 1 down-threshold 0
system-monitor LineCard threshold marginal-threshold 1 down-threshold 2
system-monitor LineCard alert state none action none
system-monitor SFM threshold marginal-threshold 1 down-threshold 2
no protocol vrrp
no protocol vrrp-extended
interface Ve 123
  shutdown
!
!
interface Management 1/0
  no ip address dhcp
  ip address 10.17.10.21/20
  ip gateway-address 10.17.0.1
  no ipv6 address autoconfig
  no ipv6 address dhcp
!
interface TenGigabitEthernet 1/0/1
  description LC 1/0/1-23
  fabric isl enable
  fabric trunk enable
  no shutdown
!
interface TenGigabitEthernet 1/0/2
  description LC 1/0/1-23
  fabric isl enable
  fabric trunk enable
  no shutdown
!
interface TenGigabitEthernet 1/0/3
  mtu 9216
  description LC 1/0/1-23
  fabric isl enable
  fabric trunk enable
  switchport
  switchport mode access
  switchport access vlan 1
  no shutdown
!
interface TenGigabitEthernet 1/0/4
  mtu 9216
  description LC 1/0/1-23
```

## Related Commands

[show global-running-config](#), [show rbridge-running config](#)

# show redundancy

Displays the control processor redundancy settings of the Management Module (MM).

## Syntax

```
show redundancy
```

## Modes

Privileged EXEC mode

## Examples

To show redundancy:

```
switch# show redundancy
=== MM Redundancy Statistics ===
Current Active Session:
Active Slot = M2 (Local), Failover Cause: Failed Over
Standby Slot = M1 (Remote)
Start Time: 11:11:08 UTC Wed Nov 28 2012
Previous Active Session:
Active Slot = M1
Standby Slot = M2
End Time: 09:50:07 UTC Wed Nov 28 2012
System Uptime: 09:42:12 UTC Wed Nov 28 2012
```

## Related Commands

[ha enable](#), [ha failover](#), [show ha](#)

# show rmon

Displays the current RMON status on the switch.

## Syntax

```
show rmon [alarms [ number ] [ brief ] | events [ number ] [ brief ] | logs [ event_number ] | statistics [ number ] [ brief ]]
```

## Parameters

### alarms

Specifies to display the RMON alarm table.

*number*

Specifies the alarm index identification number. Valid values range from 1 through 65535.

**brief**

Specifies to display a brief summary of the output.

### events

Specifies to display the RMON events table.

*number*

Specifies the event index identification number. Valid values range from 1 through 65535.

**brief**

Specifies to display a brief summary of the output.

### logs

Specifies to display the RMON log table.

*event\_number*

Specifies the event log index identification number. Valid values range from 1 through 65535.

### statistics

Specifies to display the statistics identification number.

*number*

Specifies the statistics identification number. Valid values range from 1 through 65535.

**brief**

Specifies a brief summary of the output.

## Modes

Privileged EXEC mode



## Examples

To display the RMON statistics:

```
switch# show rmon statistics

rmon collection index 4
  Interface index is Id: 67108864 , Name : Ten Gigabit Ethernet 0/0
  Receive Statistics:
    218903 packets, 14015626 bytes, 0 packs dropped
    Multicasts: 218884, Broadcasts: 18
    Under-size : 0, Jabbers: 0, CRC: 0
    Fragments: 0, Collisions: 0
      64 byte pkts: 218722, 65-127 byte pkts: 174
    128-255 byte pkts: 0, 256-511 byte pkts: 6
    512-1023 byte pkts: 0, 1024-1518 byte pkts: 0
    Over 1518-byte pkts(Oversize - Jumbo): 0
  Owner: RMON_SNMP
  Status: ok(1)
```

To display the RMON events:

```
switch# show rmon events

event Index = 4
  Description "My Description"
  Event type Log & SnmpTrap
  Event community name admin
  Last Time Sent = 00:00:00
  Owner admin
```

## Related Commands

[rmon alarm](#), [rmon event](#)

# show rmon history

Displays information gathered by rmon event and rmon alarm commands.

## Syntax

```
show rmon history [ statistics | history_index ]
```

## Parameters

### **statistics**

Displays a more detailed synopsis.

### *history\_index*

Specifies the RMON history identification number. Valid values range from 1 through 65535.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display a synopsis of the statistics collected by the **rmon event** and **rmon alarm** commands.

Add the **statistics** parameter to display the detailed history.

## Examples

To display the RMON history:

```
switch# show rmon history

RMON history control entry 1
interface: ifIndex.1745682445 Ten Gigabit Ethernet 0/13
buckets requested: 20
buckets granted: 20
sampling interval: 10
Owner: jsmith
```

## Related Commands

[rmon alarm](#), [rmon event](#)

# show route-map

Displays the PBR configuration details.

## Syntax

```
show route-map [ name ] [ rbridge-id { rbridge-id | all } ]
show route-map ve vlan-id { rbridge-id rbridge-id | all }
```

## Parameters

*name*

The name of the route-map.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

**ve** *vlan\_id*

Specifies the interface Ve for specified RBridge ID.

## Modes

Privileged EXEC mode

## Command Output

The **show port-security** command displays the following information:

Output field	Description
Active/Partial/Inactive	Indicates the instantiation of the route-map configuration into the underlying hardware. Possible meanings for inactive may be no room in the TCAM for programming the ACL, or the exhaustion of next-hop entries within the hardware next-hop table.
Selected	Indicates which of the configured next hops is currently being used by the policy. If the keyword selected is absent from the display, it indicates that none of the next hops in the list is being used and the packet is being routed by the standard routing mechanism.
Policy routing matches	Provides a summary of the number of times any of the match criteria within the specific ACL have been hit. If the ACL binding was unable to allocate a counter for the ACL (due to resource exhaustion) the count value will show "Counter not available" otherwise an actual counter value will be displayed.

## Examples

```
sw0# show route-map
Interface TenGigabitEthernet 3/3
  Ip Policy Route-map abc
Interface TenGigabitEthernet 3/4
  Ip Policy Route-map bar
```

Example of **show route-map** by application:

```
sw0# show route-map abc
Interface TenGigabitEthernet 3/3
  ip policy route-map abc permit 10 (Active)
    match ip address acl ACL_Vincent
    set ip precedence critical
    set ip next-hop 3.3.1.1 (selected)
    set ip next-hop 4.4.2.1
    Policy routing matches: 100 packets; 500000 bytes
ip policy route-map abc permit 20 (Active)
  match ip address acl ACL_Vincent_2
  set ip precedence flash
  set ip next-hop 10.3.1.1
  set ip next-hop 10.4.2.1 (selected)
  set ip interface null0
  Policy routing matches: 0 packets; 0 bytes
sw0# show route-map xyz
Interface TenGigabitEthernet 3/4
  ip policy route-map xyz deny 10 (inactive)
    match ip address acl Vincent
    set ip precedence critical
    set ip vrf pulp_fiction next-hop 3.3.3.5 (selected)
    set ip next-hop 4.4.4.4
    Policy routing matches: Counter not available
sw0# show route-map abc rbridge-id all
Interface TenGigabitEthernet 204/3/3
  ip policy route-map abc permit 10 (Active)
    match ip address acl ACL_Vincent
    set ip next-hop 3.3.1.1 (selected)
    set ip next-hop 4.4.2.1
    Policy routing matches: 100 packets; 500000 bytes
Interface Ve 3 on rbridge-id 205
  ip policy route-map abc permit 20 (Active)
    match ip address acl ACL_Vincent_2
    set ip next-hop 10.3.1.1
    set ip next-hop 10.4.2.1 (selected)
    set ip interface null0
    Policy routing matches: 0 packets; 0 bytes
```

## Related Commands

[ip policy route-map](#)

# show route-map interface

Displays the status of a route-map application on the specified interface.

## Syntax

```
show route-map interface { port-channel index | <N>gigabitethernet slot/port | ve vlan-id }
```

```
show route-map interface ve vlan-id rbridge-id { rbridge-id | all }
```

## Parameters

**port-channel** *index*

Displays the status of the port-channel interface.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

*port*

Specifies a valid port number.

**ve** *vlan-id*

Displays the status of a route-map application on the specified virtual Ethernet interface Ve for the mentioned rbridge-id.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

You do not need to specify the route map name, as only a single route map can be applied to an interface.

## Examples

To display the status of a route map on a 10-gigabit Ethernet interface:

```
sw0# show route-map interface tengigabitethernet 3/3
Interface TenGigabitEthernet 3/3
  ip policy route-map foo permit 10 (Active)
    match ip address acl ACL_Vincent
    set ip next-hop 3.3.1.1 (selected)
    set ip next-hop 4.4.2.1
    Policy routing matches: 100 packets; 500000 bytes
  ip policy route-map foo permit 20 (Active)
    match ip address acl ACL_Vincent_2
    set ip next-hop 10.3.1.1
    set ip next-hop 10.4.2.1 (selected)
    set ip interface null0
    Policy routing matches: 0 packets; 0 bytes
sw0# show route-map interface Ve 3 rbridge-id all
Interface Ve 3 on rbridge-id 205
  ip policy route-map foo permit 10 (Active)
    match ip address acl ACL_Vincent
    set ip precedence critical
    set ip next-hop 3.3.1.1 (selected)
    set ip next-hop 4.4.2.1
    Policy routing matches: 100 packets; 500000 bytes
Interface Ve 3 on rbridge-id 206
  ip policy route-map foo permit 20 (Active)
    match ip address acl ACL_Vincent_2
    set ip next-hop 10.3.1.1
    set ip next-hop 10.4.2.1 (selected)
    set ip interface null0
    Policy routing matches: 0 packets; 0 bytes
```

## Related Commands

[ip policy route-map](#)

# show running reserved-vlan

Displays the range of reserved VLAN values.

## Syntax

```
show running reserved-vlan
```

## Modes

Privileged EXEC mode

## Related Commands

[reserved-vlan](#), [show default-vlan](#)

# show running-config

Displays the contents of the running configuration.

## Syntax

**show running-config**

## Parameters

Refer to the Usage Guidelines.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the running configuration.

This command is supported only on the local device.

To display the list of available configuration entries, enter **show running-config ?**.

The **show running-config** option displays the global configuration and also the configuration on all interfaces.

The **show running-config interface** options display only the configuration for the interfaces specified.



## Examples

The following example displays the running configuration.

```
device# show running-config

chassis virtual-ip 10.24.73.50/20
no diag post enable
linecard 2 LC48x10G
linecard 4 LC48x10G
class-map default
  match any
!
logging rbridge-id 1
  raslog console INFO
!
logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
logging syslog-facility local LOG_LOCAL7
switch-attributes 1
  chassis-name VDX8770-4
  host-name sw0
!
support rbridge-id 1
  ffdc
!
snmp-server contact "Field Support."
snmp-server location "End User Premise."
snmp-server sys-descr "VDX Switch."
snmp-server community ConvergedNetwork
snmp-server community OrigEquipMfr rw
snmp-server community "Secret C0de" rw
snmp-server community common!
(Output truncated)
```

## Related Commands

[show startup-config](#), [show startup-db](#)

# show running-config aaa

Displays the configuration attributes for the authentication, authorization, and accounting (AAA) server from the configuration database.

## Syntax

```
show running-config aaa [ accounting [ commands | exec ] | authentication [ login ] ]
```

## Parameters

### accounting

Configures Login or Command accounting

### commands

Enable/Disable Command accounting

### exec

Enable/Disable Login accounting

### authentication

Configures preferred order of Authentication output modifiers

### login

Configures the order of sources for login (default = 'local')

## Modes

Privileged EXEC mode

## Usage Guidelines

Refer to the **aaa authentication** command for a description of the displayed attributes.

## Examples

To display the authentication mode:

```
device# show running-config aaa
aaa authentication radius local
aaa accounting exec default start-stop none
aaa accounting commands default start-stop none

device# show running-config aaa authentication
aaa authentication login radius local

device# show running-config aaa authentication
aaa authentication login ldap local-auth-fallback
```

## Related Commands

[aaa authentication](#)

# show running-config aaa accounting

Displays the AAA server accounting configuration.

## Syntax

```
show running-config aaa accounting
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Refer to the **aaa authentication** command for a description of the displayed attributes.

## Examples

To displaying the authentication mode:

```
device# show running-config aaa accounting
aaa accounting exec default start-stop tacacs+
aaa accounting commands default start-stop tacacs+
```

## Related Commands

[aaa authentication](#)

# show running-config ag

Displays the configured N\_Port to VF\_Port mappings, port grouping information, and other parameters for Access Gateway (AG) mode.

## Syntax

```
show running-config ag rbridge-id
```

## Command Default

Displays the AG configuration on local switch when the RBridge ID is not specified.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display parameters configured for AG mode. This shows the factory-default configuration, unless parameters have been modified by the user.

Consider the following when using LB mode with **show running-config ag** and **show ag** commands:

- The only Port Grouping mode that you can enable or disable is lb mode.
- When lb mode is disabled in a port group, the **show running-config ag**, **show ag map**, and **show ag** commands display the configured VF\_Port to N\_Port mapping. This is because configured and active mapping are the same.
- When lb mode is enabled in a port group, **show ag** and **show ag map** commands display the active mapping only because VF\_Port to N\_Port mapping is based on the current distributed load across all N\_Ports. The **show running-config ag** command displays the configured mapping only.

## Examples

The following example displays the running AG configuration for RBridge ID 1.

```
sw0# show running-config rbridge-id 1 ag
 nport 1/0/1
  map fport interface Fcoe 1/1/1 1/1/9 1/1/17 1/1/25 1/1/33 1/1/41 1/1/49 1/1/57
  !
 nport 1/0/2
  map fport interface Fcoe 1/1/2 1/1/10 1/1/18 1/1/26 1/1/34 1/1/42 1/1/50 1/1/58
  !
 nport 1/0/3
  map fport interface Fcoe 1/1/3 1/1/11 1/1/19 1/1/27 1/1/35 1/1/43 1/1/51 1/1/59
  !
 nport 1/0/4
  map fport interface Fcoe 1/1/4 1/1/12 1/1/20 1/1/28 1/1/36 1/1/44 1/1/52 1/1/60
  !
 nport 1/0/5
  map fport interface Fcoe 1/1/5 1/1/13 1/1/21 1/1/29 1/1/37 1/1/45 1/1/53 1/1/61
  !
 nport 1/0/6
  map fport interface Fcoe 1/1/6 1/1/14 1/1/22 1/1/30 1/1/38 1/1/46 1/1/54 1/1/62
  !
 nport 1/0/7
  map fport interface Fcoe 1/1/7 1/1/15 1/1/23 1/1/31 1/1/39 1/1/47 1/1/55 1/1/63
  !
 nport 1/0/8
  map fport interface Fcoe 1/1/8 1/1/16 1/1/24 1/1/32 1/1/40 1/1/48 1/1/56 1/1/64
  !
 pg 0
  nport interface FibreChannel 1/0/1 1/0/2 1/0/3 1/0/4 1/0/5 1/0/6 1/0/7 1/0/8
  modes lb
  rename pg0
  !
timeout fnm 120
counter reliability 25
!
!

sw0# show running-config ag
ag
mapset nport 1/0/1 fports 1/1/1 1/1/92 1/1/173
mapset nport 1/0/2 fports 1/1/24 1/1/105 1/1/186
mapset nport 1/0/3 fports 1/1/37 1/1/118 1/1/199
mapset nport 1/0/4 fports 1/1/410 1/1/1211 1/1/2012
mapset nport 1/0/5 fports 1/1/513 1/1/1314 1/1/2115
mapset nport 1/0/6 fports 1/1/616 1/1/1417 1/1/2218
mapset nport 1/0/7 fports 1/1/719 1/1/1520 1/1/2321
mapset nport 1/0/8 fports 1/1/822 1/1/1623 1/1/24
staticmapadd nport 1/0/1 staticfports
staticmapadd nport 1/0/2 staticfports
staticmapadd nport 1/0/3 staticfports
.....
failback 1/0/5 true
failback 1/0/6 true
failback 1/0/7 true
vcs:
fabric-map default
vlan 1002
priority 3
virtual-fabric 128
fcmap 0E:FC:00
```

## Related Commands

[show ag](#), [show ag map](#)

# show running-config banner

Displays the switch banner.

## Syntax

```
show running-config banner
```

## Modes

Privileged EXEC mode

## Examples

To display the switch banner:

```
switch# show running-config banner  
banner login "Please don't disturb the setup on this switch."
```

## Related Commands

[banner login](#)

# show running-config cee-map

Displays the Converged Enhanced Ethernet (CEE) map.

## Syntax

```
show running-config cee-map [ precedence | priority-group-table [pgid] | priority-table | remap { fabric-priority | lossless-priority } ]
```

## Parameters

### **precedence**

Displays only the precedence of the default CEE map.

### **priority-group-table**

Without a specified priority group ID, displays the priority group table for each priority group ID.

*pgid*

Specifies one priority group ID.

### **priority-table**

Displays the configured priority table map.

### **remap fabric-priority**

Displays the fabric priority for the VCS Fabric QoS.

### **remap lossless-priority**

Displays the lossless priority for the VCS Fabric QoS.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display properties of the configured CEE map. Without parameters, the command displays the precedence of the default CEE map, priority group table for each priority group ID, the configured priority table map, and the fabric priority and lossless priority for the VCS Fabric QoS.

## Examples

To display the CEE map:

```
device(config)# show running-config cee-map

cee-map default
precedence 1
priority-group-table 1 weight 40 pfc on
priority-group-table 15.0 pfc off
priority-group-table 15.1 pfc off
priority-group-table 15.2 pfc off
priority-group-table 15.3 pfc off
priority-group-table 15.4 pfc off
priority-group-table 15.5 pfc off
priority-group-table 15.6 pfc off
priority-group-table 15.7 pfc off
priority-group-table 2 weight 60 pfc off
priority-table 2 2 2 1 2 2 2 15.0
remap fabric-priority priority 0
remap lossless-priorirty priority 0
!
```

## Related Commands

[priority-group-table](#), [remap fabric-priority](#), [remap lossless-priority](#)



# show running-config class-map

Displays configured class-maps.

## Syntax

```
show running-config class-map
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is only supported on VDX 8770-4, VDX 8770-8, and later devices.

## Examples

To display configured class maps:

```
device# show running-config class-map

class-map default
match any
```

## Related Commands

[qos cos](#)

# show running-config diag post

Displays the defined POST configuration.

## Syntax

```
show running-config diag post
```

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config diag post

diag post rbridge-id 132
no enable
switch# show running-config diag post

diag post rbridge-id 132
enable
```

## Related Commands

[diag post enable](#)

# show running-config dot1x

Displays the IEEE 802.1x Port Authentication configuration.

## Syntax

```
show running-config dot1x [ enable | test timeout ]
```

## Parameters

### enable

Shows the configured state of globally enabled IEEE 802.1x port authentication.

### test timeout

Shows the configured timeout value in seconds for the IEEE 802.1x readiness check.

## Modes

Privileged EXEC mode

## Related Commands

[dot1x enable](#), [dot1x test timeout](#)

# show running-config dpod

Displays Dynamic Ports on Demand (DPOD) license information.

## Syntax

```
show running-config dpod [ rbridge-id/slot/port ]
```

## Command Default

Displays all port reservations on the local device.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display port reservations for a specified port or for all ports on the local device.

This command has no effect on VDX 8770 devices. These devices do not support the Dynamic Ports on Demand feature.

## Examples

To display port reservations for all ports on the local device:

```
device# show running-config dpod

dpod 10/0/1
  reserve
!
dpod 10/0/2
  reserve
!
dpod 10/0/3
!
dpod 10/0/4
  reserve
!
dpod 10/0/5
!
dpod 10/0/6
  reserve
!
(Output truncated)
```

To display port reservations on a device that does not support the DPOD feature:

```
device# show running-config dpod

%No entries found
```

## Related Commands

[dpod](#), [show dpod](#)

# show running-config event-handler

Displays details of one or all event-handler profiles configured on the switch. You can filter the results by description, Python-script action, or trigger ID. You can also display the Python-script action associated with a profile.

## Syntax

```
show running-config event-handler [ event-handler-name ]
show running-config event-handler event-handler-name description
show running-config event-handler event-handler-name action
show running-config event-handler event-handler-name trigger [ trigger-id { raslog raslog-id [ pattern posix-ext-regex ] | vcs } ]
```

## Parameters

*event-handler-name*

Specifies the name of the event-handler profile. Valid values can have from 1 through 32 characters. The first character must be alphabetic.

**action**

Displays by Python script file-names.

**description**

Describes the event-handler profile. The string can be 1 through 128 characters in length.

**trigger** *trigger-id*

Specifies an event-handler trigger. When the trigger-condition occurs, a Python script is run.

**raslog** *raslog-id*

Specifies a RASlog message ID as the trigger.

**pattern** *posix-ext-regex*

Specifies a POSIX extended regular expression to search for a match within the specified RASlog message ID. For examples, refer to the "trigger" topic.

**vcs**

Specifies device events as the trigger.

## Modes

Privileged EXEC mode

## Command Output

The **show running-config event-handler** command displays the following information:

Output field	Description
event-handler	Displays the event-handler name.
action python-script	Displays the name of the Python script called if the event handler is triggered.

Output field	Description
trigger	Displays a trigger name and definitions

## Examples

The following example displays the details of all triggers defined for a specified event-handler.

```
device# show running-config event-handler evh1 trigger
event-handler evh1
  trigger 10 vcs switch-ready-for-configuration
```

The following example displays the details of the action defined for a specified event-handler.

```
device# show running-config event-handler evh1 action
event-handler evh1
  action python-script vlan.py
```

The following example displays the details of all defined event-handlers.

```
device# show running-config event-handler
event-handler evh1
  trigger 10 vcs switch-ready-for-configuration
  action python-script vlan.py
!
event-handler evh2
  trigger 100 raslog NSM-1001
  action python-script vlan.py
!
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	This command was modified to support the <b>description</b> parameter. In addition, you no longer filter by parameter values.

# show running-config fabric route mcast

Displays fabric route multicast configuration information.

## Syntax

```
show running-config fabric route mcast { rbridge-id rbridge-id | priority }
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

**priority**

Displays the priority value.

## Modes

Privileged EXEC mode

## Usage Guidelines

The configuration currently effective on the switch is referred to as the running configuration. Any configuration change you make while the switch is online is made to the running configuration.

## Examples

These examples display fabric route multicast configuration information:

```
switch# show running-config fabric route mcast

fabric route mcast rbridge-id 2
switch# show running-config fabric route mcast rbridge-id 2 priority

fabric route mcast rbridge-id 2
priority 1
```

## Related Commands

[fabric route mcast](#), [show fabric route multicast](#)



# show running-config fcoe

Displays the running configuration for FCoE.

## Syntax

```
show running-config fcoe [ fabric-map default | map default ]
```

## Parameters

### **fabric-map default**

Displays the contents of the fabric map.

### **map default**

Displays the list of available maps.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config fcoe fabric-map default

fcoe
  fabric-map default
    vlan-id 1002
    priority 3
    virtual-fabric 128
    fcmap 0E:FC:00
    advertisement interval 8000
    keep-alive timeout
switch# show running-config fcoe map default

fcoe
map default
  fabric-map default
  cee-map default
```

# show running-config fcsp auth

Displays the E\_Port-to-EX\_Port authentication protocol parameters.

## Syntax

```
show running-config [ rbridge-id { rbridge-id | all } ] fcsp auth
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the E\_Port-to-EX\_Port authentication protocol parameters such as auth-type, group, hash type, and policy state.

The policy status can be one of the following: ON, OFF, ACTIVE, or PASSIVE. Refer to the **fcsp auth** command for a description of policy states.

## Examples

To display both protocol and policy (auth-type = all, group = 2, hash = md5, and switch policy = off)

```
swe52# show running-config rbridge-id 2 fcsp auth
rbridge-id 2
fcspauth auth-type all
fcspauth group 2
fcspauth hash sha1
fcspauth policy switch active
```

## Related Commands

[fcsp auth](#), [fcsp auth-secret dhchap](#), [show fcsp auth-secret dh-chap](#)

# show running-config hardware

Displays configuration information related to connector status, FlexPort configuration, custom hardware profiles, and port groups.

## Syntax

```
show running-config hardware
show running-config hardware connector [ rbridge-id/slot/port ] [ sfp [ breakout ] ]
show running-config hardware connector-group [ rbridge-id/slot/port ] [ speed ]
show running-config hardware custom-profile { kap } [ bfd-l3 | bfd-vxlan | fcoe | lacp | rpvt | udld | xstp ] [ hello-interval | num-entry ]
show running-config hardware flexport
show running-config hardware port-group
```

## Parameters

### connector

Displays the current configuration of connectors, including SFP configuration and breakout status.

*rbridge-id/slot/port*

Specifies an RBridge ID, slot, and port

### sfp

Displays Small Form-Factor Pluggable (SFP) modules.

### breakout

Displays ports in breakout mode.

### connector-group

Displays the current configuration of the connector group.

*rbridge-id/slot/port*

Specifies an RBridge ID, slot, and port

### speed

Displays connector-group speed.

### custom-profile

Displays the current configuration of custom hardware profiles.

### kap

Displays Keep-Alive Protocol (KAP) configuration.

### bfd-l3

Displays protocol KAP parameters for BFD-L3 (Bidirectional Forwarding Detection for Layer 3).

### bfd-vxlan

Displays protocol KAP parameters for BFD VXLANs.

### fcoe

Displays protocol KAP parameters for FCoE.

show running-config hardware

- lACP**  
Displays protocol KAP parameters for Link Aggregation Control Protocol.
  - rpvst**  
Displays protocol KAP parameters for Rapid Per-VLAN Spanning Tree.
  - udld**  
Displays protocol KAP parameters for Unidirectional Link Detection.
  - xstp**  
Displays protocol KAP parameters for any version of Spanning Tree Protocol.
- flexport**  
Displays the current FlexPort configuration.
- port-group**  
Displays the current port-group configuration.
- hello-interval  
Displays hello-interval settings.
  - num-entry  
Displays the number of keep-alive entries per slot.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Examples

The following example displays connector SFP and breakout information for a specified port:

```
device# show running-config hardware connector 1/0/49 sfp breakout
hardware
connector 1/0/49
no sfp breakout
!
!
```

The following example displays connector-group information with configured speed:

```
device# show running-config hardware connector-group speed
hardware
connector-group 1/0/1
  speed LowMixed
!
connector-group 1/0/3
  speed LowMixed
!
connector-group 1/0/5
  speed LowMixed
!
connector-group 1/0/6
  speed LowMixed
!
connector-group 2/0/1
  speed LowMixed
!
connector-group 2/0/3
  speed LowMixed
!
connector-group 2/0/5
  speed LowMixed
!
connector-group 2/0/6
  speed LowMixed
!
```

The following example displays custom KAP profiles and protocol settings:

```
device# show running-config hardware custom-profile
hardware
custom-profile kap mmyprofile1
  lacp hello-interval 1000
  xstp hello-interval 4000 num-entry 60
  rpvst hello-interval 2500 num-entry 100
  udld hello-interval 500 num-entry 32
  bfd-vxlan hello-interval 4000 num-entry 15
  bfd-l3 hello-interval 2000 num-entry 100
  fcoe num-entry 64
!
custom-profile kap myprofile2
!
```

The following example displays FlexPort information:

```
device# show running-config hardware flexport
hardware
flexport 1/0/1
  type ethernet
!
flexport 1/0/5
  type ethernet
!
flexport 2/0/5
  type ethernet
!
```

The following example displays port-group information:

```
device# show running-config hardware port-group
hardware
port-group 1/0/2
  mode 100g
!
port-group 1/0/3
  mode 40g
```

## History

Release version	Command history
5.0.0	This command was introduced.
6.0.1	This command was modified to add support for custom profiles.

# show running-config interface fcoe

Displays the status of FCoE interfaces.

## Syntax

```
show running-config interface fcoe [ vn-number/rbridge-id/front-port-number ]
```

## Parameters

*vn-number/rbridge-id/front-port-number*  
Specifies a valid FCoE port interface.

*vn-number*  
Specifies the VN number for FCoE.

*rbridge-id*  
Specifies an RBridge ID.

*front-port-number*  
Specifies the front port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fcoe

interface Fcoe 1/22/1
 no shutdown
!
interface Fcoe 1/22/2
 no shutdown
!
interface Fcoe 1/22/3
 no shutdown
!
interface Fcoe 1/22/4
 no shutdown
!
interface Fcoe 1/22/5
 no shutdown
!
interface Fcoe 1/22/6
 no shutdown
!
```

# show running-config interface FibreChannel

Displays Fibre Channel port attributes.

## Syntax

```
show running-config interface FibreChannel [ rbridge-id/slot/port [ isl-r_rdy | shutdown | speed | trunk-enable ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**isl-r\_rdy**

Displays whether R\_RDY buffer-to-buffer flow control is enabled for the ISL. VC\_RDY flow control is enabled if R\_RDY flow control is disabled.

**shutdown**

Displays whether the port is enabled (**no shutdown**) or disabled (**shutdown**).

**speed**

Displays the configured port speed: auto, 1 Gbps, 2 Gbps, 4 Gbps, or 8 Gbps.

**trunk-enable**

Displays whether trunking is enabled on the port.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display port attributes for a specified Fibre Channel port or for all Fibre Channel ports on the cluster.

Enter the command without specifying *rbridge-id/slot/port* to obtain a listing of attributes for all Fibre Channel ports. Enter the command with the *rbridge-id/slot/port* parameters to obtain attributes information for a specific port. The values for *isl-r\_rdy*, *trunk-enable*, and *shutdown* are always displayed. The values for *speed* are displayed only if they have been changed from their default values.

Include the attribute name to obtain the setting of that specific attribute only.

This command can be used only on Network OS platforms with Fibre Channel ports.

Enter **interface FibreChannel** to set the values.



## Examples

To display Fibre Channel port attributes for all ports on a VDX 6740 device:

```
device# show running-config interface FibreChannel

interface FibreChannel 3/0/1
  desire-distance 0
  no isl-r_rdy
  trunk-enable
  no shutdown
!
interface FibreChannel 3/0/2
  desire-distance 0
  no isl-r_rdy
  trunk-enable
  no shutdown
!
interface FibreChannel 3/0/3
  desire-distance 0
  no isl-r_rdy
  trunk-enable
  no shutdown
!
(Output truncated)
```

To display Fibre Channel port attributes for one port of a VDX 6740 device:

```
device# show running-config interface FibreChannel 8/0/1

interface FibreChannel 8/0/1
  speed 8gbps
  no isl-r_rdy
  trunk-enable
  shutdown
!
```

To display the setting of a specific attribute on a specific port:

```
device# show running-config interface FibreChannel 66/0/1 speed

interface FibreChannel 66/0/1
  speed auto
!
```

# show running-config interface fortygigabitethernet

Displays the status of 40-gigabit Ethernet interfaces.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ]
```

## Command Default

Displays the configuration of all 40-gigabit Ethernet interfaces on the local device.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

To display configuration information about all 40-gigabit Ethernet interfaces on a VDX device:

```
device# show running-config interface fortygigabitethernet

interface Forty Gigabit Ethernet 22/0/49
fabric isl enable
fabric trunk enable
no shutdown
!
interface Forty Gigabit Ethernet 22/0/50
fabric isl enable
fabric trunk enable
no shutdown
!
interface Forty Gigabit Ethernet 22/0/51
fabric isl enable
fabric trunk enable
no shutdown
!
interface Forty Gigabit Ethernet 22/0/52
fabric isl enable
fabric trunk enable
sflow enable
no shutdown
!
interface Forty Gigabit Ethernet 22/0/53
fabric isl enable
fabric trunk enable
sflow enable
shutdown
!
interface Forty Gigabit Ethernet 22/0/54
fabric isl enable
fabric trunk enable
```

## Related Commands

[interface](#)

# show running-config interface fortygigabitethernet bpdu-drop

Displays the BPDU drop status of a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] bpdu-drop [ enable ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Displays the drop status of STP/MSTP/RSTP and PVST+/R-PVST+ BPDUs.

## Modes

Privileged EXEC mode

## Usage Guidelines

STP, RSTP, or MSTP must be configured.

Network OS supports PVST+ and R-PVST+only. The PVST and R-PVST protocols are proprietary to Cisco and are not supported.

## Examples

To display BPDU drop status information for a specific 40-gigabit Ethernet port:

```
device# show running-config interface fortygigabitethernet 1/0/49 bpdu-drop
```

## Related Commands

[interface](#)

# show running-config interface fortygigabitethernet cee

Displays whether the default CEE map has been applied to a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] cee
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command does not apply to ISL ports.

## Related Commands

[cee](#)

# show running-config interface fortygigabitethernet channel-group

Displays channel group configuration information for a 40-gigabit Ethernet interface participating in link aggregation.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] channel-group [ mode | type ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**mode**

Displays the mode of link aggregation (active, passive, or on).

**type**

Displays the type of link aggregation (802.3ad standards-based LAG, or Extreme proprietary hardware-based trunking).

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is relevant only to interfaces configured as part of a LAG.

## Related Commands

[channel-group](#)

# show running-config interface fortygigabitethernet description

Displays the description string associated with a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] description
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/52 description
interface fortygigabitethernet 1/0/52
description Connects to storage device 1
```

## Related Commands

[description \(interfaces\)](#), [interface](#)

# show running-config interface fortygigabitethernet dot1x

Displays IEEE 802.1x port-based access control configuration information for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] dot1x [ authentication | port-control | protocol-version | quiet-period | reauthMax | reauthentication | timeout [ re-authperiod | server-timeout | supp-timeout | tx-period ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**authentication**

Indicates whether 802.1x port-based access control is enabled on the interface.

**port-control**

Displays the status of port authorization: auto (authentication on the port is enabled), forced authorize, or force unauthorize.

**protocol-version**

Displays the version number of the dot1x protocol.

**quiet-period**

Displays the number of seconds between a failed authentication and the next authentication retry.

**reauthMax**

Displays the maximum number of reauthentication attempts before the port goes into the reauthorized state.

**reauthentication**

Indicates whether reauthentication is enabled on a port.

**timeout**

Displays 802.1x timeout values.

**re-authperiod**

Displays the reauthentication interval in seconds.

**server-timeout**

Displays the number of seconds the switch waits for a response from the authentication server.

**supp-timeout**

Displays the number of seconds that the switch waits for a response to the Extensible Authentication Protocol (EAP) frame.



**tx-period**

Displays the number of seconds that the switch waits for a response to an EAP request or identity frame from the client before retransmitting the request

## Modes

Privileged EXEC mode

## Examples

To display the 802.1x port-based authentication configuration for a 40-gigabit Ethernet interface:

```
switch# show running-config interface fortygigabitethernet 1/0/49 dot1x

interface fortygigabitethernet 1/0/49
dot1x authentication
dot1x port-control auto
dot1x quiet-period 120
dot1x reauthMax 5
dot1x reauthentication
dot1x timeout server-timeout 60
```

# show running-config interface fortygigabitethernet fabric

Displays fabric protocol configuration parameters for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] fabric [ isl [ enable ] | trunk [ enable ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**isl [ enable ]**

Indicates only the administration and operational state the Inter-Switch Link (ISL).

**trunk [ enable ]**

Indicates only whether trunking is enabled on the port.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/49 fabric

interface fortygigabitethernet 1/0/49
 fabric isl enable
 fabric trunk enable
```

## Related Commands

[fabric isl enable](#), [fabric trunk enable](#)

# show running-config interface fortygigabitethernet fcoeport

Displays whether a 40-gigabit Ethernet interface is configured as an FCoE port.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] fcoeport
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Related Commands

[fcoe](#), [fcoeport](#)

# show running-config interface fortygigabitethernet lacp

Displays interface configuration parameters for the Link Aggregation Control Protocol (LACP) for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] lacp [ timeout ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**timeout**

Indicates whether the interface timeout is short (for Extreme trunks) or long (for standard trunks).

## Modes

Privileged EXEC mode

## Related Commands

[lacp timeout](#)

# show running-config interface fortygigabitethernet lldp

Displays Link Layer Discovery Protocol (LLDP) configuration information for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] lldp [ dcbx-version | disable | iscsi-priority | profile ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**dcbx-version**

Displays the configured version of the Data Center Bridging Exchange (DCBX) protocol.

**disable**

Indicates whether LLDP is disabled on the interface.

**iscsi-priority**

Displays the configured priority that will be advertized in the DCBX iSCSI TLV.

**profile**

Displays the LLDP profile configured on the interface.

## Modes

Privileged EXEC mode

# show running-config interface fortygigabitethernet mac

Displays configured MAC parameters for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] mac [ access-group ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**access-group**

Displays MAC ACLs configured for the specified interface.

## Modes

Privileged EXEC mode

## Related Commands

[mac access-group](#)

# show running-config interface fortygigabitethernet mtu

Displays the configured MTU for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] mtu
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/49 mtu
interface fortygigabitethernet 1/0/49
  mtu 2500
```

## Related Commands

[ip mtu](#)

# show running-config interface fortygigabitethernet port-profile-port

Displays whether AMPP port-profile configuration mode is enabled for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] port-profile-port
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/50 port-profile-port  
  
interface fortygigabitethernet 1/0/50  
port-profile-port
```

## Related Commands

[port-profile-port](#)



# show running-config interface fortygigabitethernet priority-tag

Displays whether 802.1p priority tagging is configured for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] priority-tag
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/51 priority-tag

interface fortygigabitethernet 1/0/51
 priority-tag
```

## Related Commands

[priority-tag](#)

# show running-config interface fortygigabitethernet qos

Displays the Quality of Service (QoS) configuration for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] qos [ cos | cos-mutation | cos-traffic-class |  
flowcontrol [ rx | tx ] | trust [ cos ] ]
```

## Command Default

Displays the full QoS configuration for the interface.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**cos**

Displays only the Class of Service (CoS) value configured for the interface.

**cos-mutation**

Displays the Cos-to-CoS mutation QoS map configured for the interface.

**cos-traffic-class**

Displays the CoS-to-Traffic Class QoS Map configured for the interface.

**flowcontrol**

Displays the activation status of QoS flow control on the interface.

**rx**

Displays the activation status of the receive portion of flow control for the interface.

**tx**

Displays the activation status of the transmit portion of flow control for the interface.

**trust cos**

Displays the configured QoS trust mode for the interface.

## Modes

Privileged EXEC mode

## Related Commands

[show qos flowcontrol interface](#), [show qos interface](#), [show qos queue interface](#), [show qos rcv-queue interface](#)

# show running-config interface fortygigabitethernet rmon

Displays the Remote Monitoring protocol (RMON) configuration for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] rmon [ collection [ history index | stats index ] ]
```

## Command Default

Displays all RMON collection configuration information.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**collection**

Displays configuration information for RMON collections.

**history**

Displays configuration information for RMON history collections.

*index*

Specifies a valid RMON history collection index value.

**stats**

Displays configuration information for RMON statistics collections.

*index*

Specifies a valid RMON collection control index value.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/49 rmon collection  
  
interface fortygigabitethernet 1/0/49  
  rmon collection stats 10 owner RMON_SNMP  
  rmon collection history 10 owner RMON_SNMP
```

## Related Commands

rmon collection history, rmon collection stats

# show running-config interface fortygigabitethernet sflow

Displays the sFlow configuration for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] sflow [ enable | polling-interval | sample-rate ]
```

## Command Default

Displays all sFlow configuration information for the port.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Displays whether sFlow is enabled for the port.

**polling-interval**

Displays the configured maximum number of seconds between successive samples of counters to be sent to the collector.

**sample-rate**

Displays the number of packets that are skipped before the next sample is taken for the interface.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/53 sflow

interface fortygigabitethernet 1/0/53
 sflow enable
 sflow polling-interval 10
 sflow sample-rate 100
```

## Related Commands

[sflow enable \(interface version\)](#), [sflow polling-interval \(interface version\)](#), [sflow sample-rate \(interface version\)](#)

# show running-config interface fortygigabitethernet shutdown

Displays whether a 40-gigabit Ethernet interface is enabled.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] shutdown
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/52 shutdown

interface fortygigabitethernet 1/0/52
no shutdown
```

## Related Commands

[shutdown](#)

# show running-config interface fortygigabitethernet switchport

Displays the configured switching characteristics for the 40-gigabit Ethernet Layer 2 interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id | slot | port ] switchport [ access [ vlan ] | mode | trunk  
[ allowed [ vlan ] | native-vlan | tag [ native-vlan ] ]
```

## Command Default

Displays all configured Layer 2 switching characteristics for the port.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**access**

Displays whether the Layer 2 interface is configured as access.

**access vlan**

Displays whether the specific VLAN on the Layer 2 interface is configured as access.

**mode**

Displays whether the Layer 2 interface is configured for access, trunk or converged.

**trunk**

Displays whether the Layer 2 interface is configured for trunk.

**trunk allowed**

Displays the configuration settings that determine the VLANs that will transmit and receive through the Layer 2 interface.

**trunk allowed vlan**

Displays the configuration settings for a specific VLAN.

**trunk allowed native-vlan**

Displays the configured native VLAN characteristics of the Layer 2 trunk interface for classifying untagged traffic.

**trunk tag**

Displays whether tagging is enabled.

**tag native-vlan**

Displays native VLAN tags.



## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/49 switchport

interface fortygigabitethernet 1/0/49
 switchport
 switchport mode access
 switchport access vlan 1
```

# show running-config interface fortygigabitethernet uddl

Displays Unidirectional Link Detection Protocol (UDLD) configuration information for a 40-gigabit Ethernet interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] uddl enable
```

## Command Default

This command has no defaults.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Indicates whether UDLD is enabled on the interface.

## Modes

Privileged EXEC mode

# show running-config interface fortygigabitethernet vlan

Displays information about VLAN classification groups for a 40-gigabit Ethernet Layer 2 interface.

## Syntax

```
show running-config interface fortygigabitethernet [ rbridge-id/slot/port ] vlan [ classifier [ activate [ group ] ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**classifier**

Displays VLAN classifier commands for the Layer 2 interface.

**activate**

Displays VLAN classifier activate commands for the Layer 2 interface.

**group**

Displays VLAN classifier activate group commands for the Layer 2 interface.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface fortygigabitethernet 1/0/49 vlan  
  
interface fortygigabitethernet 1/0/49  
vlan classifier activate group 1 vlan 2
```

# show running-config interface gigabitethernet

Displays the status of 1-gigabit Ethernet interfaces.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ]
```

## Command Default

Displays the configuration of all 1-gigabit Ethernet interfaces on the local switch.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

To display configuration information about all 1-gigabit Ethernet interfaces on the local switch:

```
switch# show running-config interface gigabitethernet

interface Gigabit Ethernet 22/0/1
  description tests
  channel-group 2 mode active type standard
  lacp timeout long
  sflow enable
  no shutdown
!
interface Gigabit Ethernet 22/0/2
  channel-group 2 mode active type standard
  lacp timeout long
  no shutdown
!
interface Gigabit Ethernet 22/0/3
  channel-group 2 mode active type standard
  lacp timeout long
  no shutdown
!
interface Gigabit Ethernet 22/0/4
  no shutdown
!
interface Gigabit Ethernet 22/0/5
  no shutdown
!
interface Gigabit Ethernet 22/0/6
  no shutdown
!
interface Gigabit Ethernet 22/0/7
  no shutdown
(Output truncated)
```

## Related Commands

[interface](#)

# show running-config interface gigabitethernet bpdu-drop

Displays the BPDU drop status of a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] bpdu-drop [ enable ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Displays the drop status of STP/MSTP/RSTP and PVST+/R-PVST+ BPDUs.

## Modes

Privileged EXEC mode

## Usage Guidelines

STP, RSTP, or MSTP must be configured.

Network OS supports PVST+ and R-PVST+only. The PVST and R-PVST protocols are proprietary to Cisco and are not supported.

## Examples

To display BPDU drop status information for a specific 1-gigabit Ethernet port:

```
device# show running-config interface gigabitethernet 1/0/7 bpdu-drop
```

## Related Commands

[interface](#)

# show running-config interface gigabitethernet channel-group

Displays channel group configuration information for an interface participating in link aggregation.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] channel-group [ mode | type ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**mode**

Displays the mode of link aggregation (active, passive, or on).

**type**

Displays the type of link aggregation (802.3ad standards-based LAG, or Extreme proprietary hardware-based trunking).

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is relevant only to interfaces configured as part of a LAG.

## Related Commands

[channel-group](#)

# show running-config interface gigabitethernet description

Displays the description string associated with a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] description
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/7 description
interface Gigabit Ethernet 1/0/7
description Connects to storage device 1
```

## Related Commands

[description \(interfaces\)](#), [interface](#)



# show running-config interface gigabitethernet dot1x

Displays IEEE 802.1x port-based access control configuration information for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] dot1x [ authentication | port-control | protocol-version |
quiet-period | reauthMax | reauthentication | timeout [ re-authperiod | server-timeout | supp-timeout | tx-period ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

### authentication

Indicates whether 802.1x port-based access control is enabled on the interface.

### port-control

Displays the status of port authorization: auto (authentication on the port is enabled), forced authorize, or force unauthorize.

### protocol-version

Displays the version number of the dot1x protocol.

### quiet-period

Displays the number of seconds between a failed authentication and the next authentication retry.

### reauthMax

Displays the maximum number of reauthentication attempts before the port goes into the reauthorized state.

### reauthentication

Indicates whether reauthentication is enabled on a port.

### timeout

Displays 802.1x timeout values.

### re-authperiod

Displays the reauthentication interval in seconds.

### server-timeout

Displays the number of seconds the switch waits for a response from the authentication server.

### supp-timeout

Displays the number of seconds that the switch waits for a response to the Extensible Authentication Protocol (EAP) frame.

### tx-period

Displays the number of seconds that the switch waits for a response to an EAP request or identity frame from the client before retransmitting the request

```
show running-config interface gigabitethernet dot1x
```

## Modes

Privileged EXEC mode

## Examples

To display the 802.1x port-based authentication configuration for a 1-gigabit Ethernet interface:

```
switch# show running-config interface gigabitethernet 1/0/7 dot1x

interface Gigabit Ethernet 1/0/7
 dot1x authentication
 dot1x port-control auto
 dot1x quiet-period 120
 dot1x reauthMax 5
 dot1x reauthentication
 dot1x timeout server-timeout 60
```

# show running-config interface gigabitethernet lacp

Displays interface configuration parameters for the Link Aggregation Control Protocol (LACP) for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] lacp [ timeout ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**timeout**

Indicates whether the interface timeout is short (for Extreme trunks) or long (for standard trunks).

## Modes

Privileged EXEC mode

## Related Commands

[lacp timeout](#)

# show running-config interface gigabitethernet lldp

Displays Link Layer Discovery Protocol (LLDP) configuration information for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] lldp [ dcbx-version | disable | iscsi-priority | profile ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**dcbx-version**

Displays the configured version of the Data Center Bridging Exchange (DCBX) protocol.

**disable**

Indicates whether LLDP is disabled on the interface.

**iscsi-priority**

Displays the configured priority that will be advertized in the DCBX iSCSI TLV.

**profile**

Displays the LLDP profile configured on the interface.

## Modes

Privileged EXEC mode

## Related Commands

[lldp dcbx-version](#), [lldp disable](#), [lldp iscsi-priority](#), [lldp profile](#)

# show running-config interface gigabitethernet mac

Displays configured MAC parameters for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] mac [ access-group ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**access-group**

Displays MAC ACLs configured for the specified interface.

## Modes

Privileged EXEC mode

## Related Commands

[mac access-group](#)

```
show running-config interface gigabitethernet mtu
```

# show running-config interface gigabitethernet mtu

Displays the configured MTU for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] mtu
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 mtu
interface Gigabit Ethernet 1/0/8
  mtu 2500
!
```

## Related Commands

[ip mtu](#)

# show running-config interface gigabitethernet port-profile-port

Displays whether AMPP port-profile configuration mode is enabled for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] port-profile-port
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 port-profile-port  
  
interface Gigabit Ethernet 1/0/8  
  port-profile-port
```

## Related Commands

[port-profile-port](#)

# show running-config interface gigabitethernet priority-tag

Displays whether 802.1p priority tagging is configured for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] priority-tag
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 priority-tag  
  
interface Gigabit Ethernet 1/0/8  
priority-tag
```

## Related Commands

[priority-tag](#)



# show running-config interface gigabitethernet qos

Displays the Quality of Service (QoS) configuration for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] qos [ cos | cos-mutation | cos-traffic-class | flowcontrol
[ rx | tx ] | trust [ cos ] ]
```

## Command Default

Displays the full QoS configuration for the interface.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**cos**

Displays only the Class of Service (CoS) value configured for the interface.

**cos-mutation**

Displays the Cos-to-CoS mutation QoS map configured for the interface.

**cos-traffic-class**

Displays the CoS-to-Traffic Class QoS Map configured for the interface.

**flowcontrol**

Displays the activation status of QoS flow control on the interface.

**rx**

Displays the activation status of the receive portion of flow control for the interface.

**tx**

Displays the activation status of the transmit portion of flow control for the interface.

**trust cos**

Displays the configured QoS trust mode for the interface.

## Modes

Privileged EXEC mode

# show running-config interface gigabitethernet rmon

Displays the Remote Monitoring protocol (RMON) configuration for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] rmon [ collection [ history index | stats index ] ]
```

## Command Default

Displays all RMON collection configuration information.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**collection**

Displays configuration information for RMON collections.

**history**

Displays configuration information for RMON history collections.

**stats**

Displays configuration information for RMON statistics collections.

*index*

Specifies a valid RMON collection control index value.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 rmon collection
interface Gigabit Ethernet 1/0/8
  rmon collection stats 10 owner RMON_SNMP
  rmon collection history 10 owner RMON_SNMP
```

## Related Commands

[rmon collection history](#), [rmon collection stats](#)

# show running-config interface gigabitethernet sflow

Displays the sFlow configuration for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] sflow [ enable | polling-interval | sample-rate ]
```

## Command Default

Displays all sFlow configuration information for the port.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Displays whether sFlow is enabled for the port.

**polling-interval**

Displays the configured maximum number of seconds between successive samples of counters to be sent to the collector.

**sample-rate**

Displays the number of packets that are skipped before the next sample is taken for the interface.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 sflow

interface Gigabit Ethernet 1/0/8
 sflow enable
 sflow polling-interval 10
 sflow sample-rate 100
!
```

## Related Commands

[sflow enable \(interface version\)](#), [sflow polling-interval \(interface version\)](#), [sflow sample-rate \(interface version\)](#)

# show running-config interface gigabitethernet shutdown

Displays whether a 1-gigabit Ethernet interface is enabled.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] shutdown
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 shutdown
interface Gigabit Ethernet 1/0/8
no shutdown
```

## Related Commands

[shutdown](#)

# show running-config interface gigabitethernet switchport

Displays the configured switching characteristics for the 1-gigabit Ethernet Layer 2 interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] switchport [ access [ vlan ] | mode | trunk [ allowed
[ vlan ] | native-vlan | tag [ native-vlan ] ]
```

## Command Default

Displays all configured Layer 2 switching characteristics for the port.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**access**

Displays whether the Layer 2 interface is configured as access.

**access vlan**

Displays whether the specific VLAN on the Layer 2 interface is configured as access.

**mode**

Displays whether the Layer 2 interface is configured for access, trunk or converged.

**trunk**

Displays whether the Layer 2 interface is configured for trunk.

**trunk allowed**

Displays the configuration settings that determine the VLANs that will transmit and receive through the Layer 2 interface.

**trunk allowed vlan**

Displays the configuration settings for a specific VLAN.

**trunk allowed native-vlan**

Displays the configured native VLAN characteristics of the Layer 2 trunk interface for classifying untagged traffic.

**trunk tag**

Displays whether tagging is enabled.

**tag native-vlan**

Displays tags for the native VLAN.

```
show running-config interface gigabitethernet switchport
```

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 switchport  
  
interface Gigabit Ethernet 1/0/8  
  switchport  
  switchport mode access  
  switchport access vlan 1
```

# show running-config interface gigabitethernet uddl

Displays Unidirectional Link Detection Protocol (UDLD) configuration information for a 1-gigabit Ethernet interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] uddl enable
```

## Command Default

This command has no defaults.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Indicates whether UDLD is enabled on the interface.

## Modes

Privileged EXEC mode

# show running-config interface gigabitethernet vlan

Displays information about VLAN classification groups for the 1-gigabit Ethernet Layer 2 interface.

## Syntax

```
show running-config interface gigabitethernet [ rbridge-id/slot/port ] vlan [ classifier [ activate [ group ] ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**classifier**

Displays VLAN classifier commands for the Layer 2 interface.

**activate**

Displays VLAN classifier activate commands for the Layer 2 interface.

**group**

Displays VLAN classifier activate group commands for the Layer 2 interface.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface gigabitethernet 1/0/8 vlan  
  
interface Gigabit Ethernet 1/0/8  
vlan classifier activate group 1 vlan 2
```



# show running-config interface management

Displays the management interface configuration for each interface in ascending numerical order.

## Syntax

```
show running-config interface management [ rbridge-id / port ]
show running-config interface management rbridge-id / port ip [ access-group | address [ dhcp ] | icmp [ echo-reply | rate-limiting | unreachable ] ]
show running-config interface management rbridge-id / port ipv6 [ access-group | address [ autoconfig | dhcp | eui-64 ] | icmpv6 [ echo-reply | rate-limiting | unreachable ] ]
show running-config interface management rbridge-id / port shutdown
show running-config interface management rbridge-id / port speed
show running-config interface management rbridge-id / port tcp [ burstrate | lockup ]
show running-config interface management rbridge-id / port vrf [ forwarding ]
```

## Parameters

*rbridge-id / port*

Specifies the RBridge ID—followed by a slash (/)—and the port number of a management interface. On modular switches with redundant management modules, you can configure two management ports: 1 and 2.

**ip**

Displays the IPv4 configurations for this interface.

**access-group**

Displays the IPv4/IPv6 access lists (ACLs) applied to this interface.

**address**

Displays the IPv4/IPv6 address configuration for this interface.

**dhcp**

Displays the Dynamic Host Configuration Protocol (DHCP) enablement status.

**icmp**

Displays the Internet Control Message Protocol (ICMP) controls for this interface.

**echo-reply**

Enables sending ICMP echo replies in response to echo requests.

**rate-limiting**

Enables ICMP rate limiting for incoming packet responses.

**unreachable**

Enables generation of ICMP Destination Unreachable message.

**ipv6**

Displays the IPv6 configurations for this interface.

**address**

Displays the IPv4/IPv6 address configuration for this interface.

**autoconfig**

Displays the IPv6 autoconfig status.

**dhcp**

Displays the Dynamic Host Configuration Protocol (DHCP) enablement status.

**eui-64**

Displays IPv6 addresses configured with an automatically computed EUI-64.

**icmpv6**

Displays the Internet Control Message Protocol (ICMP) controls for this interface.

**echo-reply**

Enables sending ICMP echo replies in response to echo requests.

**rate-limiting**

Enables ICMP rate limiting for incoming packet responses.

**unreachable**

Enables generation of ICMP Destination Unreachable message.

**shutdown**

Displays the shutdown status of this management interface.

**speed**

Displays the interface speed parameter.

**tcp**

Displays TCP burstrate and lockup settings.

**burstrate**

Displays the TCP burstrate setting.

**lockup**

Displays the TCP lockup setting.

**vrf**

Displays vrf status on this interface.

**forwarding**

Displays vrf forwarding status on this interface.

## Modes

Privileged EXEC mode

## Examples

The following example displays all management interface configurations.

```
device# show running-config interface management
```

```
interface Management 56/0
  no tcp burstrate
  ip icmp echo-reply
  no ip address dhcp
  ip address 10.38.19.56/20
  ipv6 icmpv6 echo-reply
  no ipv6 address autoconfig
  no ipv6 address dhcp
  vrf forwarding mgmt-vrf
  no shutdown
!
interface Management 60/0
  no tcp burstrate
  ip icmp echo-reply
  no ip address dhcp
  ip address 10.38.19.60/20
  ipv6 icmpv6 echo-reply
  no ipv6 address autoconfig
  no ipv6 address dhcp
  vrf forwarding mgmt-vrf
  no shutdown
!
interface Management 110/1
  no tcp burstrate
  ip icmp echo-reply
  no ip address dhcp
  ip address 10.38.19.110/20
  ipv6 icmpv6 echo-reply
  no ipv6 address autoconfig
  no ipv6 address dhcp
  vrf forwarding mgmt-vrf
  no shutdown
```

## Related Commands

[interface management](#), [ip access-list](#), [show running-config ip access-list](#)

# show running-config interface port-channel

Displays the status of port-channel interfaces.

## Syntax

```
show running-config interface port-channel [ number ]
```

## Command Default

Displays the configuration of all port channel interfaces on the local switch.

## Parameters

*number*  
Specifies a valid port-channel number.

## Modes

Privileged EXEC mode

## Examples

To display configuration information about all port channel interfaces on an Extreme device:

```
device# show running-config interface port-channel

interface port-channel 1
description 1
shutdown
!
interface port-channel 2
switchport
switchport mode access
switchport access vlan 1
shutdown
!
interface port-channel 3
shutdown
```

## Related Commands

[interface](#)

# show running-config interface tengigabitethernet

Displays the status of 10-gigabit Ethernet interfaces.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ]
```

## Command Default

Displays the configuration of all 10-gigabitEthernet interfaces on the local switch.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

The following example displays configuration information about all 10-gigabit Ethernet interfaces on a device.

```
device# show running-config interface tengigabitethernet

interface Ten Gigabit Ethernet 22/0/49
fabric isl enable
fabric trunk enable
no shutdown
!
interface Ten Gigabit Ethernet 22/0/50
fabric isl enable
fabric trunk enable
no shutdown
!
interface Ten Gigabit Ethernet 22/0/51
fabric isl enable
fabric trunk enable
no shutdown
!
interface Ten Gigabit Ethernet 22/0/52
fabric isl enable
fabric trunk enable
sflow enable
no shutdown
!
interface Ten Gigabit Ethernet 22/0/53
fabric isl enable
fabric trunk enable
sflow enable
shutdown
!
interface Ten Gigabit Ethernet 22/0/54
fabric isl enable
fabric trunk enable
```

The following example displays configuration information about the 10-gigabit Ethernet interfaces on a device. Both such interfaces are enabled for link-state tracking (LST).

```
device# show running-config interface tengigabitethernet
interface TenGigabitEthernet 2/0/1
  track enable
  track interface port-channel 10
  fabric isl enable
  fabric trunk enable
  switchport
  switchport mode trunk
  switchport trunk allowed vlan all
  switchport trunk tag native-vlan
  no spanning-tree shutdown
  no shutdown
!

interface TenGigabitEthernet 3/0/10
  track enable
  track interface ethernet 3/0/25
  fabric isl enable
  fabric trunk enable
  no spanning-tree shutdown
  no shutdown
```

The following example displays configuration information about a specific 10-gigabit Ethernet interface enabled for link-state tracking (LST).

```
show interface tengigabitethernet 7/0/48
TenGigabitEthernet 7/0/48 is up, line protocol is up (connected)
Hardware is Ethernet, address is 0027.f887.e140
  Current address is 0027.f887.e140
Tracking status: Enabled
Tracked interfaces:Te 7/0/47(up)  Fo 7/0/50(up)
Fixed Copper RJ45 Media Present
Interface index (ifindex) is 30266490880
MTU 2500 bytes
LineSpeed Actual      : 1000 Mbit
LineSpeed Configured : Auto, Duplex: Full
Priority Tag disable
Last clearing of show interface counters: 2d06h07m
Queueing strategy: fifo
Receive Statistics:
  0 packets, 0 bytes
  Unicasts: 0, Multicasts: 0, Broadcasts: 0
  64-byte pkts: 0, Over 64-byte pkts: 0, Over 127-byte pkts: 0
  Over 255-byte pkts: 0, Over 511-byte pkts: 0, Over 1023-byte pkts: 0
  Over 1518-byte pkts(Jumbo): 0
  Runts: 0, Jabbers: 0, CRC: 0, Overruns: 0
  Errors: 0, Discards: 0
Transmit Statistics:
  6494 packets, 811750 bytes
  Unicasts: 0, Multicasts: 6494, Broadcasts: 0
  Underruns: 0
  Errors: 0, Discards: 0
Rate info:
  Input 0.000000 Mbits/sec, 0 packets/sec, 0.00% of line-rate
  Output 0.000000 Mbits/sec, 0 packets/sec, 0.00% of line-rate
Time since last interface status change: 2d06h05m
```

The following example displays configuration information about the 10-gigabit Ethernet interface 2/2/1, enabled for RA Guard.

```
device# show running-config interface tengigabitethernent 2/2/1

interface TenGigabitEthernet 2/2/1
fabric isl enable
fabric trunk enable
switchport
switchport mode trunk
switchport trunk allowed vlan all
switchport trunk tag native-vlan
ipv6 raguard
no shutdown
!
```

## Related Commands

[interface](#)

# show running-config interface tengigabitethernet bpdu-drop

Displays the BPDU drop status of a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] bpdu-drop [ enable ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Displays the drop status of STP/MSTP/RSTP and PVST+/R-PVST+ BPDUs.

## Modes

Privileged EXEC mode

## Usage Guidelines

STP, RSTP, or MSTP must be configured.

Extreme Network OS supports PVST+ and R-PVST+ only. The PVST and R-PVST protocols are proprietary to Cisco and are not supported.

## Examples

To display BPDU drop status information for a specific 10-gigabit Ethernet port:

```
device# show running-config interface tengigabitethernet 1/0/49 bpdu-drop
```

## Related Commands

[interface](#)



# show running-config interface tengigabitethernet cee

Displays whether the default CEE map has been applied to a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] cee
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command does not apply to ISL ports.

## Related Commands

[cee](#)

# show running-config interface tengigabitethernet channel-group

Displays channel group configuration information for a 10-gigabit Ethernet interface participating in link aggregation.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] channel-group [ mode | type ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**mode**

Displays the mode of link aggregation (active, passive, or on).

**type**

Displays the type of link aggregation (802.3ad standards-based LAG, or Network OS proprietary hardware-based trunking).

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is relevant only to interfaces configured as part of a LAG.

## Related Commands

[channel-group](#)

# show running-config interface tengigabitethernet description

Displays the description string associated with a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] description
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/52 description
interface tengigabitethernet 1/0/52
description Connects to storage device 1
```

## Related Commands

[description \(interfaces\)](#), [interface](#)

# show running-config interface tengigabitethernet dot1x

Displays IEEE 802.1x port-based access control configuration information for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] dot1x [ authentication | port-control | protocol-version | quiet-period | reauthMax | reauthentication | timeout [ re-authperiod | server-timeout | supp-timeout | tx-period ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**authentication**

Indicates whether 802.1x port-based access control is enabled on the interface.

**port-control**

Displays the status of port authorization: auto (authentication on the port is enabled), forced authorize, or force unauthorize.

**protocol-version**

Displays the version number of the dot1x protocol.

**quiet-period**

Displays the number of seconds between a failed authentication and the next authentication retry.

**reauthMax**

Displays the maximum number of reauthentication attempts before the port goes into the reauthorized state.

**reauthentication**

Indicates whether reauthentication is enabled on a port.

**timeout**

Displays 802.1x timeout values.

**re-authperiod**

Displays the reauthentication interval in seconds.

**server-timeout**

Displays the number of seconds the switch waits for a response from the authentication server.

**supp-timeout**

Displays the number of seconds that the switch waits for a response to the Extensible Authentication Protocol (EAP) frame.

**tx-period**

Displays the number of seconds that the switch waits for a response to an EAP request or identity frame from the client before retransmitting the request

## Modes

Privileged EXEC mode

## Examples

To display the 802.1x port-based authentication configuration for a 10-gigabit Ethernet interface:

```
switch# show running-config interface tengigabitethernet 1/0/49 dot1x

interface tengigabitethernet 1/0/49
dot1x authentication
dot1x port-control auto
dot1x quiet-period 120
dot1x reauthMax 5
dot1x reauthentication
dot1x timeout server-timeout 60
```

# show running-config interface tengigabitethernet fabric

Displays fabric protocol configuration parameters for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] fabric [ isl [ enable ] | trunk [ enable ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**isl [ enable ]**

Indicates only the administration and operational state of the Inter-Switch Link (ISL).

**trunk [ enable ]**

Indicates only whether trunking is enabled on the port.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display ISL and trunking status for the specified 10-gigabit Ethernet interface.

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/49 fabric
interface tengigabitethernet 1/0/49
fabric isl enable
fabric trunk enable
```

## Related Commands

[fabric isl enable](#), [fabric trunk enable](#)

# show running-config interface tengigabitethernet fcoeport

Displays whether a 10-gigabit Ethernet interface is configured as an FCoE port.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] fcoeport
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Related Commands

[fcoe](#), [fcoeport](#)

# show running-config interface tengigabitethernet lacp

Displays interface configuration parameters for the Link Aggregation Control Protocol (LACP) for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] lacp [ timeout ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**timeout**

Indicates whether the interface timeout is short (for Network OS trunks) or long (for standard trunks).

## Modes

Privileged EXEC mode

## Related Commands

[lacp timeout](#)



# show running-config interface tengigabitethernet lldp

Displays Link Layer Discovery Protocol (LLDP) configuration information for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] lldp [ dcbx-version | disable | iscsi-priority | profile ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**dcbx-version**

Displays the configured version of the Data Center Bridging Exchange (DCBX) protocol.

**disable**

Indicates whether LLDP is disabled on the interface.

**iscsi-priority**

Displays the configured priority that will be advertized in the DCBX iSCSI TLV.

**profile**

Displays the LLDP profile configured on the interface.

## Modes

Privileged EXEC mode

# show running-config interface tengigabitethernet mac

Displays configured MAC parameters for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] mac [ access-group ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**access-group**

Displays MAC ACLs configured for the specified interface.

## Modes

Privileged EXEC mode

## Related Commands

[mac access-group](#)

# show running-config interface tengigabitethernet mtu

Displays the configured MTU for a 10 gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] mtu
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/49 mtu
interface tengigabitethernet 1/0/49
  mtu 2500
```

## Related Commands

[ip mtu](#)

# show running-config interface tengigabitethernet port-profile-port

Displays whether AMPP port-profile configuration mode is enabled for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] port-profile-port
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/50 port-profile-port  
  
interface tengigabitethernet 1/0/50  
port-profile-port
```

## Related Commands

[port-profile-port](#)

# show running-config interface tengigabitethernet priority-tag

Displays whether 802.1p priority tagging is configured for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] priority-tag
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/51 priority-tag  
  
interface tengigabitethernet 1/0/51  
priority-tag
```

## Related Commands

[priority-tag](#)

# show running-config interface tengigabitethernet qos

Displays the quality of service (QoS) configured for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] qos [ cos | cos-mutation | cos-traffic-class |  
flowcontrol [ rx | tx ] | trust [ cos ] ]
```

## Command Default

Displays the full QoS configuration for the interface.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**cos**

Displays only the Class of Service (CoS) value configured for the interface.

**cos-mutation**

Displays the Cos-to-CoS mutation QoS map configured for the interface.

**cos-traffic-class**

Displays the CoS-to-Traffic Class QoS Map configured for the interface.

**flowcontrol**

Displays the activation status of QoS flow control on the interface.

**rx**

Displays the activation status of the receive portion of flow control for the interface.

**tx**

Displays the activation status of the transmit portion of flow control for the interface.

**trust cos**

Displays the configured QoS trust mode for the interface.

## Modes

Privileged EXEC mode

# show running-config interface tengigabitethernet rmon

Displays the Remote Monitoring protocol (RMON) configuration for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] rmon [ collection [ history index | stats index ] ]
```

## Command Default

Displays all RMON collection configuration information.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**collection**

Displays configuration information for RMON collections.

**history**

Displays configuration information for RMON history collections.

*index*

Specifies a valid RMON history collection index value.

**stats**

Displays configuration information for RMON statistics collections.

*index*

Specifies a valid RMON collection control index value.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/49 rmon collection

interface tengigabitethernet 1/0/49
 rmon collection stats 10 owner RMON_SNMP
 rmon collection history 10 owner RMON_SNMP
```

show running-config interface tengigabitethernet rmon

## Related Commands

[rmon collection history](#), [rmon collection stats](#)



# show running-config interface tengigabitethernet sflow

Displays the sFlow configuration for a 10-gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] sflow [ enable | polling-interval | sample-rate ]
```

## Command Default

Displays all sFlow configuration information for the port.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Displays whether sFlow is enabled for the port.

**polling-interval**

Displays the configured maximum number of seconds between successive samples of counters to be sent to the collector.

**sample-rate**

Displays the number of packets that are skipped before the next sample is taken for the interface.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/53 sflow

interface tengigabitethernet 1/0/53
 sflow enable
 sflow polling-interval 10
 sflow sample-rate 100
```

## Related Commands

[sflow enable \(interface version\)](#), [sflow polling-interval \(interface version\)](#), [sflow sample-rate \(interface version\)](#)

# show running-config interface tengigabitethernet shutdown

Displays whether a 10-gigabit Ethernet interface is enabled.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] shutdown
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/52 shutdown  
  
interface tengigabitethernet 1/0/52  
no shutdown
```

## Related Commands

[shutdown](#)

# show running-config interface tengigabitethernet switchport

Displays the configured switching characteristics for the 10-gigabit Ethernet Layer 2 interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id | slot | port ] switchport [ access [ vlan ] | mode | trunk [ allowed
[ vlan ] | native-vlan | tag [ native-vlan ] ]
```

## Command Default

Displays all configured Layer 2 switching characteristics for the port.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**access**

Displays whether the Layer 2 interface is configured as access.

**access vlan**

Displays whether the specific VLAN on the Layer 2 interface is configured as access.

**mode**

Displays whether the Layer 2 interface is configured for access, trunk or converged.

**trunk**

Displays whether the Layer 2 interface is configured for trunk.

**trunk allowed**

Displays the configuration settings that determine the VLANs that will transmit and receive through the Layer 2 interface.

**trunk allowed vlan**

Displays the configuration settings for a specific VLAN.

**trunk allowed native-vlan**

Displays the configured native VLAN characteristics of the Layer 2 trunk interface for classifying untagged traffic.

**trunk tag**

Displays whether tagging is enabled.

**tag native-vlan**

Displays tags for the native VLAN.

```
show running-config interface tengigabitethernet switchport
```

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/49 switchport  
  
interface tengigabitethernet 1/0/49  
  switchport  
  switchport mode access  
  switchport access vlan 1
```

# show running-config interface tengigabitethernet uddl

Displays Unidirectional Link Detection Protocol (UDLD) configuration information for a 10 Gigabit Ethernet interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] uddl enable
```

## Command Default

This command has no defaults.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**enable**

Indicates whether UDLD is enabled on the interface.

## Modes

Privileged EXEC mode

# show running-config interface tengigabitethernet vlan

Displays information about VLAN classification groups for a 10-gigabit Ethernet Layer 2 interface.

## Syntax

```
show running-config interface tengigabitethernet [ rbridge-id/slot/port ] vlan [ classifier [ activate [ group ] ] ]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**classifier**

Displays VLAN classifier commands for the Layer 2 interface.

**activate**

Displays VLAN classifier activate commands for the Layer 2 interface.

**group**

Displays VLAN classifier activate group commands for the Layer 2 interface.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config interface tengigabitethernet 1/0/49 vlan  
  
interface tengigabitethernet 1/0/49  
vlan classifier activate group 1 vlan 2
```

# show running-config interface vlan

Displays the status of VLAN interfaces.

## Syntax

```
show running-config interface vlan [ vlan_id ] [ arp-ageing-timeout | description | ip | mac access-group | shutdown | spanning-tree ]
```

## Command Default

Displays the configuration of all VLAN interfaces on the local switch.

## Parameters

*vlan\_id*

Specifies a VLAN.

**arp-ageing-timeout**

Displays the configured interface timeout value in minutes for the Address Resolution Protocol (ARP) for VLANs.

**description**

Displays the description text entered for each VLAN or for the specified VLAN.

**ip**

Displays IP configuration information for VLANs.

**mac access-group**

Displays MAC ACLs configured for VLANs.

**shutdown**

Specifies whether the VLAN interface is enabled.

**spanning-tree**

Displays spanning tree configuration information for VLANs.

## Modes

Privileged EXEC mode

## Related Commands

[interface](#), [show running-config interface vlan ip](#)

# show running-config interface vlan ip

Displays the IP configuration of VLAN interfaces.

## Syntax

```
show running-config interface vlan [ vlan_id ] ip [ address | igmp [ last-member-query-interval | query-interval | query-max-response-time | snooping [ enable | fast-leave | mrouter | mrouter-timeout | querier ] | static-group static-group-address ] | mtu | proxy-arp ]
```

## Command Default

Displays configured information for all VLAN interfaces on the local switch.

## Parameters

*vlan\_id*

Specifies a VLAN.

**address**

Displays the IP address configured for VLANs.

**igmp**

Displays whether the Internet Group Management Protocol (IGMP) is enabled for VLANs.

**last-member-query-interval**

Displays the amount of time in seconds that the IGMP router waits to receive a response to a group query message.

**query-interval**

Displays the amount of time in seconds between IGMP query messages sent by the switch.

**query-max-response-time**

Displays the configured maximum response time in seconds for IGMP queries.

**snooping**

Displays IGMP snooping configuration information for VLANs.

**enable**

Indicates whether IGMP snooping is enabled for specified VLANs.

**fast-leave**

Indicates if snooping fast leave is enabled.

**mrouter**

Displays multicast router port information for the VLAN.

**mrouter-timeout**

Displays the configured multicast router IGMP timeout value in seconds.

**querier**

Indicates if IGMP snooping querier is configured.

**static-group**

Displays configured static group membership entries.



***static-group-address***

Specifies an IPv4 address to return static group information about.

**mtu**

Displays the MTU configured for each VLAN.

**proxy-arp**

Indicates whether a proxy ARP is configured for VLAN interfaces.

## Modes

Privileged EXEC mode

## Examples

To display IP configuration information for all configured VLANs:

```
switch# show running-config interface vlan ip

interface Vlan 1
!
interface Vlan 2
ip igmp query-interval 200
ip igmp query-max-response-time 15
ip igmp snooping enable
```

## Related Commands

[interface management](#), [ip access-list](#), [show running-config interface management](#)

# show running-config ip access-list

Displays a list of IPv4 ACLs defined on the switch, including the rules they contain.

## Syntax

```
show running-config ip access-list [ { standard | extended } [ ACL_name ] ]
```

## Parameters

### standard

Specifies the standard ACL type.

### extended

Specifies the extended ACL type.

### ACL\_name

Specifies the ACL name.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

Not specifying **standard** or **extended** displays a list of all IPv4 ACLs defined on the switch.

If you specify **standard** or **extended**, you can also specify an ACL.

To display details of IPv4 ACLs bound to interfaces, use the **show access-list ip** command.

## Examples

The following example displays the IPv4 ACLs defined on the switch.

```
device# show running-config ip access-list

ip access-list standard stdACL3
  seq 5 permit host 10.20.33.4
  seq 7 permit any
ip access-list extended extdACL5
  seq 5 deny tcp host 10.24.26.145 any eq 23
  seq 7 deny tcp any any eq 80
  seq 10 deny udp any any range 10 25
  seq 15 permit tcp any
ip access-list extended extdACLwithNoRules
```

# show running-config ipv6 access-list

Displays a list of IPv6 ACLs defined on the switch, including the rules they contain.

## Syntax

```
show running-config ipv6 access-list [ { standard | extended } [ ACL_name ] ]
```

## Parameters

### standard

Specifies the standard ACL type.

### extended

Specifies the extended ACL type.

### ACL\_name

Specifies the ACL name.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

Not specifying **standard** or **extended** displays a list of all IPv6 ACLs defined on the switch.

If you specify **standard** or **extended**, you can also specify an ACL.

To display details of all IPv6 ACLs bound to interfaces, use the **show access-list ipv6** command.

## Examples

The following example displays all standard IPv6 ACLs defined on the switch:

```
device# show running-config ipv6 access-list standard
ipv6 access-list standard distList
  seq 10 deny 2001:125:132:35::/64
  seq 20 deny 2001:54:131::/64
  seq 30 deny 2001:5409:2004::/64
  seq 40 permit any
!
ipv6 access-list standard ipv6_acl_std_1
  seq 10 deny 2001:2001::/64 count log
```

# show running-config ip dns

Displays the domain name service (DNS) parameters. The DNS parameters are the domain name and the name server IP address for primary and secondary name servers.

## Syntax

```
show running-config ip dns
```

## Modes

Privileged EXEC mode

## Examples

To display the configured DNS parameters:

```
device# show running-config ip dns

ip dns domain-name extremenetworks.com
ip dns name-server 10.70.20.1
ip dns name-server 10.70.20.10
```

## Related Commands

[ip dns](#)

# show running-config ip igmp

Displays IGMP configuration information.

## Syntax

```
show running-config ip igmp [ snooping [ enable ] ]
```

## Parameters

### snooping

Displays IGMP snooping configuration information.

### enable

Displays whether IGMP snooping is enabled.

## Modes

Privileged EXEC mode

## Examples

To display IGMP configuration information:

```
switch# show running-config ip igmp
```

## Related Commands

[ip igmp snooping enable](#), [ip igmp snooping enable \(global version\)](#)

# show running-config ip route

Displays routing information.

## Syntax

```
show running-config ip route [ routing-table ]
```

## Parameters

**routing-table**

Displays a specific route to a specific destination.

## Modes

Privileged EXEC mode

# show running-config ldap-server

Displays the LDAP server status in the running-config.

## Syntax

```
show running-config ldap-server [ host ipaddr | host-name ]
```

## Parameters

### host

Identifies the IPv4 address of the host.

### *ipaddress*

IPv4 address of the host.

### host-name

Name of the host.

## Modes

Privileged EXEC mode

## Usage Guidelines

LDAP server configuration is placed at the beginning of the running-config and is part of the global configuration of the switch. LDAP is enabled by default and no entry is shown in the running-config when set to default.

Attributes with default values will not be displayed.

## Examples

```
device# show running-config ldap-server host 10.24.65.6

ldap-server host 10.24.65.6
  port          3890
  domain        security.extremenetworks.com
  retries       3
!
switch#
```

## Related Commands

[certutil import ldapca](#), [ldap-server host](#), [ldap-server maprole](#)

# show running-config line

Displays command line session configuration information.

## Syntax

```
show running-config line [ vty [ exec-timeout ] ]
```

## Parameters

**vty**

Displays the terminal type.

**exec-timeout**

Displays the configured idle time in minutes before the command line session automatically logs off.

## Modes

Privileged EXEC mode

## Related Commands

[interface](#)



# show running-config logging

Displays the configuration of the logging facilities on the local switch.

## Syntax

```
show running-config logging
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

## Examples

To display the logging facilities configured on the local switch:

```
device# show running-config logging
logging raslog console INFO
logging syslog-server 1.1.1.1 use-vrf mgmt-vrf
!
logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
logging syslog-facility local LOG_LOCAL0
logging syslog-client localip CHASSIS_IP
device#
```

## History

Release version	Command history
6.0.1	This command output was modified.

## Related Commands

[show running-config logging auditlog class](#), [show running-config logging syslog-server](#)

# show running-config logging auditlog class

Displays the severity level configured for the audit log class.

## Syntax

```
show running-config logging auditlog class
```

## Command Default

Displays the information for the local switch.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

This command is not supported on the standby management module.

## Examples

To display audit log classes enabled on the switch:

```
switch# show running-config logging auditlog class

logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
```

## Related Commands

[clear logging auditlog](#), [clear logging raslog](#)

# show running-config logging raslog

Displays the severity level configured for the RASLog console.

## Syntax

```
show running-config logging raslog
```

## Command Default

Displays the RASLog console configuration.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the configured severity levels for the RASlog console. Valid values consist of one of the following: INFO, WARNING, ERROR, or CRITICAL.

This command is supported only on the local switch.

This command is not supported on the standby management module.

## Examples

To display the severity level configured for the RASlog console:

```
switch# show running-config logging raslog
logging raslog console INFO
```

## Related Commands

[clear logging raslog](#), [logging raslog console](#), [show running-config logging](#)

# show running-config logging syslog-client

Displays the syslog client configuration.

## Syntax

```
show running-config logging syslog-client
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the clients that are running the syslogd daemon and to which system messages are sent. Clients are specified in the configuration database by IP address.

This command is supported only on the local switch.

This command is not supported on the standby management module.

## Examples

To display the syslog client configuration:

```
device# show running-config logging syslog-client
logging syslog-client localip CHASSIS_IP
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show running-config logging syslog-facility

Displays the syslog facility log level.

## Syntax

```
show running-config logging syslog-facility [ local ]
```

## Command Default

Displays the local configuration.

## Parameters

**local**

Displays the local syslog facility level.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

This command is not supported on the standby management module.

## Examples

To display the syslog daemon IP addresses configured on a switch:

```
switch# show running-config logging syslog-facility
logging syslog-facility local LOG_LOCAL7
```

## Related Commands

[logging syslog-server](#), [show running-config logging syslog-server](#)

# show running-config logging syslog-server

Displays the syslog server configuration.

## Syntax

```
show running-config logging syslog-server
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the servers that are running the syslogd daemon and to which system messages are sent. Servers are specified in the configuration database by IP address.

This command is supported only on the local switch.

This command is not supported on the standby management module.

## Examples

To display the syslog daemon IP addresses configured on a switch:

```
switch# show running-config logging syslog-server

logging syslog-server 10.17.17.203
  secure port 6514
!
logging syslog-server 10.17.17.204
```

## Related Commands

[logging syslog-server](#)

# show running-config mac access-list

Displays a list of MAC ACLs defined on the switch, including the rules they contain.

## Syntax

```
show running-config mac access-list [ { standard | extended } [ ACL_name ] ]
```

## Parameters

### standard

Specifies the standard ACL type.

### extended

Specifies the extended ACL type.

### ACL\_name

Specifies the ACL name.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

Not specifying **standard** or **extended** displays a list of all MAC ACLs defined on the switch.

If you specify **standard** or **extended**, you can also specify an ACL.

To display details of all MAC ACLs bound to interfaces, use the **show access-list mac** command.

## Examples

The following example displays all MAC ACLs defined on the switch.

```
device# show running-config mac access-list
mac access-list standard stdmacaclin
seq 11 permit 1111.1112.1113 7777.7777.7777 count log
seq 12 permit 1111.1112.1114 7777.7777.7777 count log
```

# show running-config mac-address-table

Displays configuration information about MAC interfaces and configurations.

## Syntax

```
show running-config monitor mac-address-table [ aging-time | static ]
```

## Command Default

Default aging time is 300 seconds.

## Parameters

### aging-time

Specifies the aging time value (in seconds).

### static

Specifies a static MAC address.

## Modes

Privileged EXEC mode



# show running-config monitor

Displays configuration information about the monitor session.

## Syntax

```
show running-config monitor { session session_number { description } }
```

## Parameters

**session** *session\_number*

The session number to display.

**description**

Displays the session's description.

## Modes

Privileged EXEC mode

## Examples

To display the monitor information:

```
switch# show running-config monitor  
monitor session 22
```

show running-config nas server-ip

# show running-config nas server-ip

Displays information about the specified Auto NAS (automatic network attached storage) interface.

## Syntax

`show running-config nas server-ip`

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on Network OS VDX 8770-4, VDX 8770-8, VDX 6740, and VDX 6740T devices.

# show running-config ntp

Displays the Network Time Protocol (NTP) server configuration.

## Syntax

```
show running-config ntp
```

## Modes

Privileged EXEC mode

## Examples

To display the configured NTP server

```
switch# show running-config ntp  
ntp server 172.26.1.159
```

## Related Commands

[ntp server](#), [show ntp status](#)

# show running-config ntp authentication-key

Displays the currently configured authentication key for accessing the NTP server.

## Syntax

```
show running-config ntp authentication-key
```

## Modes

Privileged EXEC mode

## Examples

Typical command output:

```
device# show running-config ntp authentication-key
ntp authentication-key 10 sha1 "cXGFY75bvKpJruCYEiwjw==\n"encryption-level 7
```

## History

Release version	Command history
5.0.2	This command was introduced.

# show running-config openflow-controller

Displays the running configuration of the OpenFlow controller.

## Syntax

```
show running-config openflow controller [name [ ip [ address | method | port ] ]
```

## Parameters

*name*

Name of an OpenFlow controller. Alphanumeric characters, hyphens, and underscores are allowed.

ip

Specifies the IPv4 address configuration of the OpenFlow controller.

address

Specifies the IPv4 address.

method

Specifies the connection method.

port

Specifies the OpenFlow controller port number.

## Modes

Privileged EXEC mode

## Examples

The following example displays the running configuration of the OpenFlow controller.

```
device# show running-config openflow-controller
openflow-controller BVC
  ip address 10.24.82.10 port 6653
!
```

## History

Release version	Command history
6.0.1	This command was introduced.

# show running-config overlay-gateway

Displays the configured characteristics for an overlay gateway.

## Syntax

```
show running-config overlay-gateway [ gateway_name ]
```

## Parameters

*gateway\_name*

Specifies the overlay-gateway name. Since only one overlay-gateway per device is supported, you do not need to enter this parameter. The default name is "gateway1".

## Modes

Privileged EXEC mode

## Command Output

The **show running-config overlay-gateway** command displays the following information:

Output field	Description
mac access-group	Specifies the name of a Layer 2 access control list (ACL) applied to the overlay gateway.
ip access-group	Specifies the name of an IPv4 ACL applied to the overlay gateway.
ipv6 access-group	Specifies the name of an IPv6 ACL applied to the overlay gateway.

## Examples

The following example displays the overlay-gateway running configuration for an overlay gateway that includes multiple sites.

```
device# show running-config overlay-gateway
overlay-gateway gw121
type layer2-extension
ip interface Loopback 11
attach rbridge-id add 1-2
map vlan 1 vni 5001
map vlan 10 vni 5010
map vlan 2 vni 5002
map vlan 3 vni 5003
map vlan 4 vni 5004
map vlan 5 vni 5005
map vlan 6 vni 5006
map vlan 7 vni 5007
map vlan 8 vni 5008
map vlan 9 vni 5009
site b
  ip address 2.2.2.2
  extend vlan add 3-4
!
site br
  ip address 3.3.3.3
  extend vlan add 5-6
!
site mu
  ip address 4.4.4.4
  extend vlan add 7-10
!
site san
  ip address 1.1.1.1
  extend vlan add 1-2
!
enable statistics direction both vlan add 5-10
monitor session 11 direction both remote-endpoint any vlan add 5-10
sflow sflowprofile1 remote-endpoint any vlan add 5-10
mac access-group stdmacaclin in
ip access-group stdipaclin in
ipv6 access-group stdipv6aclin in
```

## Related Commands

[overlay-gateway](#), [show overlay-gateway](#)

# show running-config ovldb-server

Displays the status of Open vSwitch Database (OVSDB) SSL servers in the running configuration.

## Syntax

`show running-config ovldb-server`

## Modes

Privileged EXEC mode

## Examples

To view the status (in this case, activated) of the OVSDB SSL servers.

### NOTE

Currently only one server can be configured.

```
device# show running-config ovldb-server
ovldb-server myserver
activate
!
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show running-config password-attributes

Displays global password attributes.

## Syntax

```
show running-config password-attributes [ admin-lockout ] [ max-lockout-duration ] [ max-retry ] [ min-length ]  
show running-config password-attributes character-restriction [ lower | numeric | special-char | upper ]
```

## Parameters

### admin-lockout

Displays lockout for admin role accounts.

### max-retry

Displays the number of failed password logins permitted before a user is locked out. Values range from 0 through 16 attempted logins. The default value is 0.

### min-length

Displays the minimum length of the password. Valid values range from 8 through 32 characters. The default is 8 characters.

### max-lockout-duration

Displays the maximum number of minutes after which the user account is unlocked. Range is from 0 through 99999. The default is 0, representing an infinite duration.

### character-restriction

Displays the restriction on various types of characters.

#### lower

Displays the minimum number of lowercase alphabetic characters that must occur in the password. Values range from 0 through 32 characters. The default value is 0.

#### numeric

Displays the minimum number of numeric characters that must occur in the password. Values range from 0 through 32 characters. The default is 0.

#### special-char

Displays the number of punctuation characters that must occur in the password. All printable, nonalphanumeric punctuation characters, except colon (:) are allowed. Values range from 0 through 32 characters. The default value is 0.

#### upper

Displays the minimum number of uppercase alphabetic characters that must occur in the password. Values range from 0 through 32 characters. The default value is 0.

## Modes

Privileged EXEC mode

## Usage Guidelines

The attributes are not displayed when they hold default values.

## Examples

The following example displays all global password attributes.

```
device# show running-config password-attributes

password-attributes max-retry 4
password-attributes character-restriction upper 1
password-attributes character-restriction lower 2
password-attributes character-restriction numeric 1
password-attributes character-restriction special-char 1
password-attributes max-lockout-duration 5000
```

## Related Commands

[service password-encryption](#)

# show running-config police-priority-map

Displays configured police-priority-maps.

## Syntax

```
show running-config class-map
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is only supported on Extreme VDX 8770-4, VDX 8770-8, and later devices.

## Examples

To display configured police-priority-maps:

```
device# show running-config police-priority-map

police-priority-map pmap1
conform 0 1 1 2 1 2 1 1
exceed 3 3 3 3 4 5 6 7
```

## Related Commands

[police-priority-map](#)

# show running-config policy-map

Displays the currently running policy-map configurations.

## Syntax

`show running-config policy-map`

## Modes

Privileged EXEC mode

## Usage Guidelines

Output includes the policy-map name, class-map name, and class-map configuration.

## Examples

To currently running policy-maps and their configuration:

```
switch# show running-config policy-map

policy-map policy_map1
  class default
    police cir 50000 cbs 500000 eir 60000 ebs 40000 set-priority prio_map1 conform-set-dscp 23 conform-
set-tc 4 exceed-set-prec 2 exceed-set-tc 5
  !
!
policy-map policy_map2
  class default
    police cir 1000000 cbs 200000
```

# show running-config port-profile

Displays configured AMPP port-profiles.

## Syntax

```
show running-config port-profile [ name ]
```

## Parameters

*name*

Specifies the name of a port-profile. If no name is provided, information about all port-profiles is displayed.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config port-profile

port-profile default
vlan-profile
switchport
switchport mode trunk
switchport trunk allowed vlan all
switchport trunk native-vlan 1
```

# show running-config port-profile activate

Displays activated AMPP port-profiles.

## Syntax

```
show running-config port-profile [ name ] activate
```

## Parameters

*name*

Specifies the name of a port-profile. If no name is provided, information about all activated port-profiles is displayed.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command display port profiles that are activated. These port profiles are available for association with MAC addresses.

## Related Commands

[port-profile \(global configuration mode\)](#), [show running-config port-profile](#)

# show running-config port-profile fcoe-profile

Displays the configured FCOE subprofile.

## Syntax

```
show running-config port-profile [ name ] fcoe-profile [ fcoeport [ default ] ]
```

## Parameters

*name*

Specifies the name of a port-profile. If no name is provided, information about all port-profiles with the FCOE profile applied is displayed.

**fcoeport [ default ]**

Specifies an FCOE map name. The only map name supported currently is "default."

## Modes

Privileged EXEC mode

# show running-config port-profile qos-profile

Displays the configured Quality of Service (QoS) subprofile.

## Syntax

```
show running-config port-profile [ name ] qos-profile [ cee [ name ] | qos [ cos cos | cos-mutation name | cos-traffic-class name | flowcontrol [ pfc | rx | tx ] | trust [ cos ] ]
```

## Parameters

*name*

Specifies the name of a port-profile. If no name is provided, information about all port-profiles with the QoS subprofile applied is displayed.

**cee** [ *name* ]

The configured QoS CEE map.

**qos**

The QoS profile.

**cos** *cos*

The configured default class of service (CoS).

**cos-mutation** *name*

The applied Cos-to-Cos mutation map.

**cos-traffic-class** *name*

The applied Cos-to-Traffic class map.

**flowcontrol**

The configured IEEE 802.3x flow control.

**pfc**

Whether priority-based flow control (PFC) is enabled.

**rx**

Whether Pause reception is enabled.

**tx**

Whether Pause generation is enabled.

**trust**

The configured QoS trust configuration.

**cos**

Whether the Layer 2 CoS field in incoming packets is configured as trusted for deriving the internal traffic class.

## Modes

Privileged EXEC mode



# show running-config port-profile security-profile

Displays security subprofiles.

## Syntax

```
show running-config port-profile [ name ] security-profile [ mac [ access-group [ acl-name | in ] ] ]
```

## Parameters

*name*

Specifies the name of a port-profile. If no name is provided, information about all port-profiles with the security subprofile applied is displayed.

**mac**

The configured MAC parameters.

**access-group**

The applied ACL.

*acl-name*

Specifies an ACL.

**in**

Ingress direction.

## Modes

Privileged EXEC mode

# show running-config port-profile static

Displays statically associated VM MAC addresses and the port profiles with which they are statically associated.

## Syntax

```
show running-config port-profile [ name ] static [ mac-address ]
```

## Parameters

*name*

Specifies the name of a port-profile. If no name is provided, information about all port-profiles associated with VM MAC addresses is displayed.

*mac-address*

Displays the port-profile associated with a specific MAC address.

## Modes

Privileged EXEC mode

## Related Commands

[port-profile \(global configuration mode\)](#), [show running-config port-profile](#)

# show running-config port-profile vlan-profile

Displays information about VLAN subprofiles.

## Syntax

```
show running-config port-profile [ name ] vlan-profile [ switchport [ access [ vlan [ vlan_id ] ] | mode [ access | trunk ] | trunk
[ allowed [ vlan [ add [ vlan_id ] ] | all | except vlan_id | none | remove [ vlan_id ] ] | native-vlan vlan_id ] ] ]
```

## Parameters

*name*

Specifies the name of a port-profile. If no name is provided, information about all port-profiles with a VLAN subprofile applied is displayed.

**switchport**

Specifies the configured switching characteristics of the Layer 2 interfaces.

**access**

Specifies VLAN interfaces for which access is configured for the VLAN profile mode.

**vlan** *vlan\_id*

Specifies a VLAN interface configured for access.

**mode**

Specifies the configured mode of the Layer 2 interface.

**access**

Specifies Layer 2 interfaces configured for access mode.

**trunk**

Specifies Layer 2 interfaces configured for trunk mode.

**trunk**

Specifies Layer 2 interfaces configured for trunk mode.

**allowed**

Specifies VLANs that are configured to transmit and receive through the Layer 2 interface.

**vlan add** [ *vlan\_id* ]

Specifies VLANs that are allowed to transmit and receive through the Layer 2 interface.

**vlan all**

Specifies all VLANs that are allowed to transmit and receive through the Layer 2 interface.

**vlan except** *vlan\_id*

Specifies VLANs that are excluded from transmitting and receiving through the Layer 2 interface.

**vlan none**

Specifies VLANs that are allowed to transmit and receive through the Layer 2 interface.

**vlan remove** [ *vlan\_id* ]

Specifies VLANs to be removed from those allowed to transmit and receive through the Layer 2 interface.

show running-config port-profile vlan-profile

**native-vlan** *vlan\_id*

Specifies native VLANs configured to classify untagged traffic

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config port-profile vlan-profile

port-profile default
vlan-profile
switchport
switchport mode trunk
switchport trunk allowed vlan all
switchport trunk native-vlan 1
!
!
switch# show running-config port-profile vlan-profile switchport trunk native-vlan

port-profile default
vlan-profile
switchport trunk native-vlan 1
!
!
```

# show running-config port-profile-domain

Displays the port-profile domains and their associated port-profiles.

## Syntax

```
show running-config port-profile-domain
```

## Modes

Privileged EXEC mode

## Usage Guidelines

```
switch# show running-config port-profile-domain
port-profile-domain PP0
  port-profile ppl
  port-profile pp4
!
port-profile-domain PP1
  port-profile pp3
  port-profile pp4
```

# show running-config preprovision

Shows the currently configured RBridge-ID-to-WWN mapping. A switch that uses this mapping to join a logical chassis cluster is called a bare-metal switch.

## Syntax

```
show running-config preprovision [ rbridge-id rbridge-id ]
```

## Parameters

### rbridge-id

Displays the RBridge ID currently being used for preprovisioning a bare-metal switch.

## Modes

Privileged EXEC mode.

## Command Output

The **show running-config preprovision** command displays the following information:

Output field	Description
RBridge ID	RBridge ID associated with the specified WWN.
WWN	WWN of the bare-metal switch..

## Examples

To view the currently configured RBridge-ID-to\_WWN mapping:

```
switch# show running-config preprovision
preprovision
  rbridge-id 5
  wwn 10:11:12:13:14:15:16:17
```

## History

Release version	Command history
6.0.0	This command was introduced.

# show running-config protocol cdp

Displays the Cisco Discovery Protocol (CDP) information.

## Syntax

```
show running-config protocol cdp
```

## Modes

Privileged EXEC mode

# show running-config protocol edge

Displays the Edge Loop Detection (ELD) parameters.

## Syntax

```
show running-config protocol edge { hello-interval | pdu-rx-limit | shutdown-time }
```

## Parameters

### hello-interval

Displays the hello-interval-limit value.

### pdu-rx-limit

Displays the bpdu-rx-limit value.

### shutdown-time

Displays the shutdown-time-limit value.

## Modes

Privileged EXEC mode



# show running-config protocol lldp

Displays the Link Layer Discovery Protocol (LLDP) parameters.

## Syntax

```
show running-config protocol lldp advertise { { dcbx-fcoe-app-tlv | dcbx-fcoe-logical-link-tlv | dcbx-iscsi-app-tlv | dcbx-tlv |
dot1-tlv | dot3-tlv | optional-tlv } | description | disable | hello | iscsi-priority | mode | multiplier | profile { description } |
system-description | system-name }
```

## Parameters

### advertise

Displays the Advertise TLV configuration information.

### dcbx-fcoe-app-tlv

Displays the IEEE Data Center Bridging eXchange FCoE Application TLV information.

### dcbx-fcoe-logical-link-tlv

Displays the IEEE Data Center Bridging eXchange FCoE Logical Link TLV information.

### dcbx-iscsi-app-tlv

Displays the IEEE Data Center Bridging eXchange iSCSI Application TLV information.

### dcbx-tlv

Displays the IEEE Data Center Bridging eXchange TLV information.

### dot1-tlv

Displays the IEEE 802.1 Organizationally Specific TLV information.

### dot3-tlv

Displays the IEEE 802.3 Organizationally Specific TLV information.

### optional-tlv

Displays the Optional TLVs information.

### description

Displays the User description

### disable

Displays the Disable LLDP

### hello

Displays the Hello Transmit interval.

### iscsi-priority

Displays the Ethernet priority to advertise for iSCSI

### mode

Displays the LLDP mode.

### multiplier

Displays the Timeout Multiplier

### profile description

Displays the LLDP Profile table and description.

show running-config protocol lldp

**system-description**

Displays the System Description.

**system-name**

Displays the System Name

## Modes

Privileged EXEC mode

# show running-config protocol spanning-tree mstp

Displays the protocol configuration information for MSTP.

## Syntax

```
show running-config protocol spanning-tree mstp [ bridge-priority | cisco-interopability | description | error-disable-  
timeout | forward-delay | instance | max-age | max-hops | port-channel | region | revision | information | shutdown |  
transmit-holdcount | vlan ]
```

## Parameters

### bridge-priority

Displays the Bridge priority commands.

### cisco-interopability

Displays the Cisco Interoperability status.

### description

Displays the spanning tree description.

### error-disable-timeout

Displays the Error-disable-timeout for the spanning tree.

### forward-delay

Displays the forward delay for the spanning tree.

### hello-time

Displays the hello time settings.

### instance

Displays the MST instance.

### max-age

Displays the max age for the spanning tree.

### max-hops

Displays the MST max hop count.

### port-channel

Displays the status of port-channel for spanning-tree.

### region

Displays the MST region.

### revision

Displays the revision number for configuration information.

### shutdown

Displays the status of the spanning-tree protocol.

### transmit-holdcount

Displays the current transmit hold count of the bridge.

### vlan

Displays the VLAN ID

show running-config protocol spanning-tree mstp

## Modes

Privileged EXEC mode

## Related Commands

[spanning-tree shutdown](#)

# show running-config protocol spanning-tree pvst

Displays the protocol configuration information for PVST.

## Syntax

```
show running-config protocol spanning-tree pvst [ bridge-priority | cisco-interopability | description | error-disable-timeout | forward-delay | instance | max-age | max-hops | port-channel | region | revision | information | shutdown | transmit-holdcount | vlan ]
```

## Parameters

### bridge-priority

Displays the Bridge priority commands.

### description

Displays the spanning tree description.

### error-disable-timeout

Displays the Error-disable-timeout for the spanning tree.

### forward-delay

Displays the forward delay for the spanning tree.

### hello-time

Displays the hello time settings.

### max-age

Displays the max age for the spanning tree.

### port-channel

Displays the status of port-channel for spanning-tree.

### shutdown

Displays the status of the spanning-tree protocol.

### vlan

Displays the VLAN ID

## Modes

Privileged EXEC mode

## Related Commands

[spanning-tree shutdown](#)

# show running-config protocol spanning-tree rpvst

Displays the protocol configuration information for RPVST.

## Syntax

```
show running-config protocol spanning-tree rpvst [ bridge-priority | cisco-interopability | description | error-disable-  
timeout | forward-delay | instance | max-age | max-hops | port-channel | region | revision | information | shutdown |  
transmit-holdcount | vlan ]
```

## Parameters

### bridge-priority

Displays the Bridge priority commands.

### cisco-interopability

Displays the Cisco Interoperability status.

### description

Displays the spanning tree description.

### error-disable-timeout

Displays the Error-disable-timeout for the spanning tree.

### forward-delay

Displays the forward delay for the spanning tree.

### hello-time

Displays the hello time settings.

### max-age

Displays the max age for the spanning tree.

### port-channel

Displays the status of port-channel for spanning-tree.

### shutdown

Displays the status of the spanning-tree protocol.

### transmit-holdcount

Displays the current transmit hold count of the bridge.

### vlan

Displays the VLAN ID

## Modes

Privileged EXEC mode

## Related Commands

[spanning-tree shutdown](#)

# show running-config protocol spanning-tree rstp

Displays the protocol configuration information for RSTP.

## Syntax

```
show running-config protocol spanning-tree rstp [ bridge-priority | cisco-interopability | description | error-disable-  
timeout | forward-delay | instance | max-age | max-hops | port-channel | region | revision | information | shutdown |  
transmit-holdcount | vlan ]
```

## Parameters

### bridge-priority

Displays the Bridge priority commands.

### description

Displays the spanning tree description.

### error-disable-timeout

Displays the Error-disable-timeout for the spanning tree.

### forward-delay

Displays the forward delay for the spanning tree.

### hello-time

Displays the hello time settings.

### max-age

Displays the max age for the spanning tree.

### port-channel

Displays the status of port-channel for spanning-tree.

### shutdown

Displays the status of the spanning-tree protocol.

### transmit-holdcount

Displays the current transmit hold count of the bridge.

## Modes

Privileged EXEC mode

## Related Commands

[spanning-tree shutdown](#)

# show running-config protocol spanning-tree stp

Displays the protocol configuration information for STP.

## Syntax

```
show running-config protocol spanning-tree stp [ bridge-priority | cisco-interopability | description | error-disable-timeout  
| forward-delay | instance | max-age | max-hops | port-channel | region | revision | information | shutdown | transmit-  
holdcount | vlan ]
```

## Parameters

### bridge-priority

Displays the Bridge priority commands.

### description

Displays the spanning tree description.

### error-disable-timeout

Displays the Error-disable-timeout for the spanning tree.

### forward-delay

Displays the forward delay for the spanning tree.

### hello-time

Displays the hello time settings.

### max-age

Displays the max age for the spanning tree.

### port-channel

Displays the status of port-channel for spanning-tree.

### shutdown

Displays the status of the spanning-tree protocol.

## Modes

Privileged EXEC mode

## Related Commands

[spanning-tree shutdown](#)



# show running-config protocol udd

Displays the UDLD global parameters.

## Syntax

```
show running-config protocol udd advertise { hello | multiplier | shutdown }
```

## Command Default

This command has no defaults.

## Parameters

### hello

Displays the Hello Transmit interval.

### multiplier

Displays the Timeout Multiplier.

### shutdown

Displays the shutdown status.

## Modes

Privileged EXEC mode

# show running-config radius-server

Displays the local device configuration for the RADIUS server from the configuration database.

## Syntax

```
show running-config radius-server host { ip-address | hostname }
```

## Parameters

### host

Identifies the RADIUS server by host name or IP address.

### *hostname*

Specifies the host name of the RADIUS server.

### *ip-address*

Specifies the IP address of the RADIUS server. IPv4 and IPv6 are supported.

## Modes

Privileged EXEC mode

## Examples

```
device# show running-config radius-server host 10.38.37.180

radius-server host 10.38.37.180
protocol    pap
key         changedsec
timeout     3
```

## Related Commands

[radius-server](#), [show running-config tacacs-server](#), [tacacs-server](#)

# show running-config rbridge-id

Displays configuration for the RBridge ID.

## Syntax

```
show running-config rbridge-id rbridge-id [fcoe]
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

**fcoe**

Displays the FCoE information for the RBridge ID.

## Modes

Privileged EXEC mode

## Examples

```
switch# show running-config rbridge-id 2

rbridge-id 2
 interface-nodespecific ns-vlan 10
 interface-nodespecific ns-ethernet 100
 fabric vlag 10 load-balance src-dst-mac-vid
 fabric vlag 20 load-balance dst-mac-vid
 no protocol vrrp

switch# show running-config rbridge-id 21 fcoe
rbridge-id 21
 fcoe
  fcoe-enodes 1000
```

## Related Commands

[show fabric ecmp load-balance](#)

# show running-config rbridge-id crypto

Displays the running configuration for HTTPS security configuration.

## Syntax

```
show running-config rbridge-id crypto [ ca | key ]
```

## Parameters

**ca**

Displays running config crypto trustpoint and associated key pair information.

**key**

Displays running config crypto key pair information.

## Modes

Privileged EXEC mode

## Usage Guidelines

To execute this command from other configuration modes, use the **do** command modifier.

## Examples

Typical command display output.

```
switch# show running-config rbridge-id crypto
rbridge-id 1
crypto key label k1 rsa modulus 2048
crypto ca trustpoint t1
  keypair k1
```

## History

Release version	Command history
6.0.0	This command was introduced.

# show running-config rbridge-id dhcpd enable

Displays the DHCP server enable status in the running configuration.

## Syntax

```
show running-config rbridge rbridge-id dhcpd enable
```

## Parameters

*rbridge-id*

Specifies the rbridge number.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Examples

The following example displays the running configuration on rbridge 1.

```
device# show running-config rbridge-id 1 dhcpd enable
rbridge-id 1
dhcpd enable
!
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show running-config rbridge-id event-handler

Displays the event-handlers configured for one or all RBridge IDs. You can limit the results to activated event-handlers and filter by additional parameters.

## Syntax

```
show running-config rbridge-id [ rbridge-id ] event-handler [ activate ]
show running-config rbridge-id [ rbridge-id ] event-handler activate event-handler-name [ action-timeout | delay | interval |
iterations | run-mode | time-window | trigger-function | trigger-mode ]
```

## Parameters

### *rbridge-id*

Specifies an RBridge. If you do not specify an RBridge, results are displayed for all RBridges.

### **activate**

Specifies activated event-handlers.

### *event-handler-name*

Specifies an event-handler profile. Valid values can have from 1 through 32 characters. The first character must be alphabetic.

### **action-timeout**

Filters by the number of minutes from when an action-script begins until execution is aborted.

### **delay**

Filters by the number of seconds from when a trigger is received until the execution of the specified action begins.

### **interval**

Filters by the number of seconds between iterations of an event-handler action, if triggered.

### **iterations**

Filters by the number of times an event-handler action is run, when triggered.

### **run-mode**

Filters by if a triggered event-handler action is run in exclusive or non-exclusive mode.

#### **exclusive**

From the triggering of an event-handler action through the completion of the action, cluster formation is not allowed to run. Exclusive run-mode can be applied to configuration-based action scripts where the script can run to completion without cluster formation interrupting configuration NOSCLIs. The exclusive run-mode will hold-off cluster formation operations such as new nodes joining the cluster or existing nodes rejoining the cluster. An active exclusive run-mode will not prevent cluster fail-over operations when the principal node itself is offline and isolated from the cluster and a new principal node is selected.

#### **non-exclusive**

Cluster formation can occur while a triggered action is in progress.

### **action-timeout**

Filters by the number of minutes from when an action-script begins until execution is aborted.

### **delay**

Filters by the number of seconds from when a trigger is received until the execution of the specified action begins.

**interval**

Specifies the number of seconds between iterations of an event-handler action, if triggered.

**iterations**

Filters by the number of times an event-handler action is run, when triggered.

**run-mode**

Filters by if a triggered event-handler action is run in exclusive or non-exclusive mode.

**exclusive**

From the triggering of an event-handler action through the completion of the action, cluster formation is not allowed to run. Exclusive run-mode can be applied to configuration-based action scripts where the script can run to completion without cluster formation interrupting configuration NOSCLIs. The exclusive run-mode will hold-off cluster formation operations such as new nodes joining the cluster or existing nodes rejoining the cluster. An active exclusive run-mode will not prevent cluster fail-over operations when the principal node itself is offline and isolated from the cluster and a new principal node is selected.

**non-exclusive**

Cluster formation can occur while a triggered action is in progress.

**time-window**

Filters by the time window within which all of the triggers must occur in order that the event-handler action runs. Following an initial triggering of an event-handler action, any subsequent trigger launches the action an additional time if the following conditions are true:

- The **trigger-mode** parameter is set to the default **each-instance**.
- The subsequent trigger occurs within the specified **time-window**.

**trigger-function**

Filters—if multiple triggers are defined for an event-handler action—if the action runs only if all of the triggers occur (**AND**) or if one is sufficient (**OR**).

**trigger-mode**

Filters by if an event-handler action can be triggered only once or more than once.

**each-instance**

The event-handler action is launched on each trigger instance received.

**on-first-instance**

As long as the switch is running, the event-handler action is launched only once. Following a switch restart, the event-handler action can be triggered again.

**only-once**

For the duration of a switch's configuration, the event-handler action is launched only once.

## Modes

Privileged EXEC mode

## Usage Guidelines

For explanations of the output fields, refer to the parameter descriptions.

## Examples

The following example displays the details of all event-handlers activated on RBridge 1.

```
device# show running-config rbridge-id 1 event-handler
rbridge-id 1
  event-handler activate evh1
    delay 10
    iterations 3
    interval 5
    run-mode exclusive
    trigger-mode only-once
    action-timeout 1800
  !
!
```

The following example displays the details of a specified event-handler, if it is activated on RBridge 1.

```
device# show running-config rbridge-id 1 event-handler activate evh1
rbridge-id 1
  event-handler activate evh1
    delay 10
    iterations 3
    interval 5
    run-mode exclusive
    trigger-mode only-once
    action-timeout 1800
  !
!
```

The following example displays the **delay** definition of a specified event-handler, if it is activated on RBridge 1.

```
device# show running-config rbridge-id 1 event-handler activate evh1 delay
rbridge-id 1
  event-handler activate evh1
    delay 10
  !
!
```

The following example displays the **action-timeout** definition of a specified event-handler, if it is activated on RBridge 1.

```
device# show running-config rbridge-id 1 event-handler activate evh1 action-timeout
rbridge-id 1
  event-handler activate evh1
    action-timeout 1800
  !
!
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	This command was modified to support the action-timeout parameter. In addition, you no longer filter by parameter values.



# show running-config rbridge-id hardware-profile

Displays the Keep-Alive Protocol (KAP), route table, and TCAM profiles in the running configuration for all RBridge IDs or a specified RBridge ID.

## Syntax

```
show running-config rbridge-id [ rbridge-id ] hardware-profile kap [ custom-profile ]
```

```
show running-config rbridge-id [ rbridge-id ] hardware-profile route-table [ maximum_paths | openflow ]
```

```
show running-config rbridge-id [ rbridge-id ] hardware-profile tcam
```

## Parameters

*rbridge-id*

Specifies an RBridge. If you do not specify an RBridge, results are displayed for all RBridges.

**kap**

Displays Keep-Alive Protocol (KAP) profiles.

**custom-profile**

Displays custom KAP profiles.

**route-table**

Displays hardware resources for route-table profiles.

*maximum\_paths*

Displays the configured number of maximum load-sharing paths.

*openflow*

Displays the status of OpenFlow support.

**tcam**

Displays hardware resources for TCAM profiles.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Examples

### Examples

The following shows the use of the **show running-config rbridge-id hardware-profile** command, without the specification of an RBridge ID, to display the results of the configuration on all RBridge IDs.

```
device# show running-config rbridge-id hardware-profile
rbridge-id 2
 hardware-profile tcam ipv4-v6-mcast maximum_paths 16 openflow on
 hardware-profile route-table ipv6-max-nd
 hardware-profile kap myprofile
!
rbridge-id 12
 hardware-profile tcam l2-ipv4-acl
 hardware-profile route-table ipv4-max-arp maximum_paths 8 openflow off
 hardware-profile kap default
```

## History

Release version	Command history
6.0.1	This command was modified to include support for the Keep-Alive Protocol (KAP), maximum paths, and OpenFlow.

# show running-config rbridge-id interface

Displays the status of interfaces for the specified RBridge.

## Syntax

```
show running-config rbridge-id rbridge-id interface [ loopback port-number | ve vlan_id
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

**interface**

Specifies an interface.

**loopback** *port-number*

Specifies a loopback interface and the port number for the loopback interface. The range is 1 through 255.

**ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface and the VLAN ID of the interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command must be executed in the current RBridge ID context.

## Examples

This example displays configuration information for loopback interface 1 for the specified RBridge.

```
device# show running-config rbridge-id 109 interface loopback 1
rbridge-id 109
 interface Loopback 1
  no shutdown
  ipv6 address 1000:9::9/128
  ip address 9.0.0.9/32
!
```

This example displays configuration information for all VE interfaces for the specified RBridge.

```
device# show running-config rbridge-id 109 interface ve
rbridge-id 109
interface Ve 20
  vrf forwarding vrf2
  ipv6 address use-link-local-only
  no shutdown
!
interface Ve 22
  vrf forwarding vrf2
  ipv6 anycast-address 2:22:1::254/64
  ip anycast-address 2.22.1.254/24
  no shutdown
!
interface Ve 23
  vrf forwarding vrf2
  ipv6 anycast-address 2:23:1::254/64
  ip anycast-address 2.23.1.254/24
  no shutdown
!
interface Ve 24
  vrf forwarding vrf2
  ipv6 anycast-address 2:24:1::254/64
  ip anycast-address 2.24.1.254/24
  no shutdown
!
interface Ve 25
  vrf forwarding vrf2
  ipv6 anycast-address 2:25:1::254/64
  ip anycast-address 2.25.1.254/24
  no shutdown
!
interface Ve 26
  vrf forwarding vrf2
  ipv6 anycast-address 2:26:1::254/64
  ip anycast-address 2.26.1.254/24
  no shutdown
...
```

# show running-config rbridge-id linecard

Displays the line card configuration.

## Syntax

```
show running-config rbridge-id rbridge-id linecard
```

## Parameters

*rbridge-id*  
Specifies an RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command must be executed in the current RBridge ID context.

## Examples

To display the line card configuration for the local switch:

```
switch# show running-config rbridge-id 1 linecard

rbridge-id 1
linecard 1 LC48x10G
linecard 2 LC48x10G
linecard 3 LC12x40G
linecard 4 LC48x10G
```

## Related Commands

[linecard](#)

# show running-config rbridge-id maps

Displays current MAPS configuration information for the specified RBridge ID.

## Syntax

```
show running-config rbridge-id [rbridge_id] [maps | email | [enable [actions| policy policy_name]] | rule rule_name | relay domainname string | group [members | type [sfp | interface]]]
```

## Parameters

**rbridge-id** *rbridge\_id*

Specifies an RBridge ID for which to display the MAPS statistics.

**email**

Displays the current email setting for the RBRidge ID.

**enable**

Displays the currently enabled actions and policies for the RBRidge ID.

**policy**

Displays the currently enabled policies for the RBRidge ID.

**actions**

Displays the currently enabled actions for the RBRidge ID.

**relay domainname** *string*

Displays the current relay setting for the designated domain name.

**group**

Displays the current configured logical groups.

**members**

Displays the members of groups in the current configured logical groups.

**type**

Displays the type of groups in the current configured logical groups.

**sfp**

Displays SFP information of the current configured logical groups.

**interface**

Displays the interface information of the current configured logical groups.

**rule** *rule\_name*

Displays the current configured rules. The valid completions for the rule names are group, interval, monitor, op, and value.

**policy**

Displays the current configured policy. The valid completion for the policy name is rule.

## Modes

Privileged EXEC mode

## Examples

Typical command example:

```
device(config-rbridge-id-5-maps)# do show running-config rbridge-id 1 maps
rbridge-id 1
maps
enable policy dflt_aggressive_policy
```

## History

Release version	Command history
6.0.1	This command was introduced.
7.0.0	This command was modified to add the parameters <b>group</b> , <b>rule</b> , and <b>policy</b> .

# show running-config rbridge-id openflow

Displays the running configuration of the OpenFlow controller by RBridge ID.

## Syntax

```
show running-config rbridge-id openflow [ logical-instance [instance-ID { activate | controller | default-behavior | passive |
version } ] ]
```

## Parameters

### logical-instance

Name of an OpenFlow controller. Alphanumeric characters, hyphens, and underscores are allowed.

#### *instance-ID*

Filters by ID of the logical instance.

#### active

Filters by activated logical instance.

#### controller

Filters by OpenFlow controller name.

#### default-behavior

Filters by default MISS behavior.

#### passive

Filters by passive controller connection.

#### version

Filters by OpenFlow version.

## Modes

Privileged EXEC mode

## Examples

The following example displays the basic output of the **show running-config rbridge-id openflow** command.

```
device# show running-config rbridge-id openflow
rbridge-id 1
openflow logical-instance 1
  version ofv130
  controller BVC
  activate
!
```

## History

Release version	Command history
6.0.1	This command was introduced.



# show running-config rbridge-id ssh

Displays the Secure Shell (SSH) configuration for an RBridge ID.

## Syntax

```
show running-config rbridge-id rbridge-id ssh
```

## Parameters

*rbridge-id*

Specifies an RBridge ID.

## Modes

Privileged EXEC mode

# show running-config rbridge-id ssh server

Displays the maximum number of Secure Shell (SSH) sessions configured on an RBridge, as well as server key types.

## Syntax

```
show running-config rbridge-id [ rbridge-id { rbridge-id | all } ] ssh server
```

## Parameters

- rbridge-id**  
Specifies an RBridge or all RBridges.
  - rbridge-id*  
Specifies an RBridge ID.
  - all**  
Specifies all RBridges.
- ssh server**  
Specifies SSH server status.

## Modes

Privileged EXEC mode

## Examples

To display the maximum number of SSH sessions configured, as well as server key types, on RBridge 176:

```
device# show running-config rbridge-id ssh server
rbridge-id 176
ssh server max-sessions 7
ssh server key rsa 2048
ssh server key ecdsa 256
ssh server key dsa
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show running-config rmon

Displays Remote Monitor configuration information.

## Syntax

```
show running-config rmon [ alarm | event ]
```

## Parameters

### alarm

Displays the Remote Monitor alarm configuration.

### event

Displays the Remote Monitor event configuration

## Modes

Privileged EXEC mode

## Related Commands

[rmon alarm](#), [rmon event](#)

# show running-config role

Displays name and description of the configured roles.

## Syntax

```
show running-config role [ name role_name [ desc ] ]
```

## Parameters

**name** *role\_name*

Displays roles defined for users.

**desc**

Displays role descriptions.

## Modes

Privileged EXEC mode

## Examples

The following example displays all roles configured on the device.

```
device# show running-config role

role name VLANAdmin desc "Manages security CLIs"
role name NetworkAdmin desc "Manages Network CLIs"
role name ClusterAdmin desc "Manages Cluster CLIs"
```

## Related Commands

[role name](#), [rule](#), [show running-config rule](#)

# show running-config route-map

Displays the status of a route-map application on the specified interface.

## Syntax

```
show running-config route-map [ name ]
```

## Parameters

*name*

Specifies the name of the route-map.

## Modes

Privileged EXEC mode

## Usage Guidelines

There is no need to specify the route map name as the user is only allowed to apply a single route map to an interface.

## Examples

```
sw0# show running-config route-map abc
ip policy route-map abc permit 20
match ip address acl Vincent
set ip vrf pulp_fiction next-hop 3.3.3.5
set ip next-hop 4.4.4.4switch#
```

## Related Commands

[route-map](#)

# show running-config rule

Displays configured access rules.

## Syntax

```
show running-config rule [ index ]
```

```
show running-config rule index { action | command command_name | operation | role }
```

```
show running-config rule { action { reject | accept } | command command_name | operation { read-only | read-write } | role role-name }
```

## Parameters

*index*

Displays the rule with the specified index number. Values range from 1 through 512.

**action reject | accept**

Following the *index* parameter, indicates whether **reject** or **accept** is specified for that rule. If the *index* parameter is not specified, displays all rules with the specified action.

**command** *command\_name*

Displays rule configuration for the specified command. To display a list of supported commands, type a question mark (?). This list varies according to whether or not you specify a rule index.

**operation read-only | read-write**

Following the *index* parameter, indicates whether **read-only** or **read-write** is specified for that rule. If the *index* parameter is not specified, displays all rules with the specified operation.

**role** *role-name*

Displays rule configuration for the specified role.

## Modes

Privileged EXEC mode

## Examples

The following example displays the configured roles and their rules.

```
device# show running-config rule

rule 30 action accept operation read-write role NetworkSecurityAdmin
rule 30 command role
!
rule 31 action accept operation read-write role NetworkSecurityAdmin
rule 31 command rule
!
rule 32 action accept operation read-write role NetworkSecurityAdmin
rule 32 command username
!
rule 33 action accept operation read-write role NetworkSecurityAdmin
rule 33 command aaa
!
rule 34 action accept operation read-write role NetworkSecurityAdmin
rule 34 command radius-server
!
rule 35 action accept operation read-write role NetworkSecurityAdmin
rule 35 command configure
!
rule 40 action accept operation read-write role FCOEAdmin
rule 40 command "interface fcoe"
```

The following example displays a single rule.

```
device# show running-config rule 30

rule 30
action accept operation read-write role NetworkSecurityAdmin command role
```

## Related Commands

[role name, rule, show running-config role](#)

# show running-config secpolicy

Displays the Switch Connection Control (SCC) security policy information.

## Syntax

```
show running-config [ rbridge-id { rbridge-id | all } ] secpolicy { defined-policy | active-policy }
```

## Parameters

### defined-policy

Displays the defined policy and its policy member set.

### active-policy

Displays the active policy and its policy member set.

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the active policy and the defined policy and its policy member set.

## Examples

To show only the defined policy of rbridge-id 3:

```
device# show running-config rbridge-id 3 secpolicy defined-policy
rbridge-id 3
secpolicy defined-policy SCC_POLICY
member-entry aa:aa:aa:aa:aa:aa:aa:aa
!
member-entry bb:bb:bb:bb:bb:bb:bb:bb
!
member-entry cc:cc:cc:cc:cc:cc:cc:cc
!
switch#
```



To show only the active policy of rbridge-id 3:

```
device# show running-config rbridge-id 3 secpolicy active-policy
rbridge-id 3
secpolicy active-policy SCC_POLICY
member-entry aa:aa:aa:aa:aa:aa:aa:aa
!
member-entry bb:bb:bb:bb:bb:bb:bb:bb
!
member-entry cc:cc:cc:cc:cc:cc:cc:cc
!
!
switch#
```

To show both active and defined policies of rbridge-id 3:

```
device# show running-config rbridge-id 3 secpolicy
rbridge-id 3
secpolicy defined-policy SCC_POLICY
member-entry aa:aa:aa:aa:aa:aa:aa:aa
!
member-entry bb:bb:bb:bb:bb:bb:bb:bb
!
member-entry cc:cc:cc:cc:cc:cc:cc:cc
!
!
secpolicy active-policy SCC_POLICY
member-entry aa:aa:aa:aa:aa:aa:aa:aa
!
member-entry bb:bb:bb:bb:bb:bb:bb:bb
!
member-entry cc:cc:cc:cc:cc:cc:cc:cc
```

## Related Commands

[secpolicy activate](#), [secpolicy defined-policy](#)

# show running-config sflow

Displays the IPv4 and IPv6 addresses and ports of sFlow collectors.

## Syntax

**show running-config sflow**

## Modes

Privileged EXEC mode

## Examples

To display the IPv4 and IPv6 addresses and ports of sFlow collectors:

```
device# show running-config sflow
sflow enable
sflow collector 1.1.1.1 50 use-vrf ""
sflow collector 1.1.1.1 50 use-vrf mgmt-vrf
sflow collector 1.1.1.1 55 use-vrf ""
sflow collector 10.10.10.10 55 use-vrf mgmt-vrf
sflow collector 2004:384::21:22 55 use-vrf ""
sflow source-ip mm-ip
device#
```

## History

Release version	Command history
6.0.1	The command output was modified.

# show running-config sflow-policy

Displays the configured sFlow policies.

## Syntax

```
show running-config sflow-policy
```

## Modes

Privileged EXEC mode

## Examples

To display the configured sFlow policies.

```
switch# show running-config sflow-policy
```

show running-config sflow-profile

# show running-config sflow-profile

Displays the configured sFlow policies.

## Syntax

`show running-config sflow-profile`

## Modes

Privileged EXEC mode

# show running-config snmp-server

Shows the user-configured running configuration of the SNMP server on the device.

## Syntax

```
show running-config snmp-server
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the current SNMP configurations of host, community, contact, user, and location.

This command has no default configurations.

## Examples

The following command shows the running configuration of the SNMP server on the device, with encryption applied:

```
device# show running-config snmp-server

snmp-server contact "Field Support."
snmp-server location "End User Premise."
snmp-server sys-descr "VDX device."
snmp-server community ConvergedNetwork
snmp-server community OrigEquipMfr groupname rw
snmp-server community "Secret C0de" groupname rw
snmp-server community common
snmp-server community private groupname rw
snmp-server community public
snmp-server host 10.17.37.107 public
use-vrf mgmt-vrf
snmp-server user snmp
snmp-server user snmpadmin1 groupname snmpadmin auth md5 auth-password "MVb+360X3kcfBzug5Vo6dQ==\n"
priv DES priv-password "ckJFoHbzVvhR0xFRPjsMTA==\n" encrypted
snmp-server user snmpadmin2 groupname snmpadmin auth md5 auth-password "MVb+360X3kcfBzug5Vo6dQ==\n"
priv DES priv-password "ckJFoHbzVvhR0xFRPjsMTA==\n" encrypted
snmp-server user snmpadmin3 groupname snmpadmin
snmp-server user snmpuser2
snmp-server user snmpuser3 auth md5 auth-password "MVb+360X3kcfBzug5Vo6dQ==\n" priv DES priv-password
"ckJFoHbzVvhR0xFRPjsMTA==\n" encrypted
```

## History

Release version	Command history
6.0.1	This command example was modified to include "use-vrf" support.

# show running-config snmp-server context

Displays SNMP contexts and their mappings to VRF instances.

## Syntax

```
show running-config snmp-server context
```

## Modes

Privileged EXEC mode

## Examples

To display an SNMP context name and its mapping to a VRF instance:

```
device# show running-config snmp-server context
snmp-server context mycontext
vrf myvrf
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show running-config snmp-server engineid

Shows the user-configured engine ID of the SNMP server on the switch.

## Syntax

```
show running-config snmp-server engineid
```

## Modes

Privileged EXEC mode

## Examples

To see the engine ID of the SNMP server:

```
switch# show running-config rbridge-id 1 snmp-server engineID local 10:20:30:40:50:60:70:80:90:10:30:12
```

# show running-config snmp-server mib community-map

Displays the mapping between and SNMP community string and a context.

## Syntax

```
show running-config snmp-server mib community-map
```

## Modes

Privileged EXEC mode

## Examples

To display the mapping between and SNMP community string and a context:

```
device# show running-config snmp-server mib community-map
snmp-server mib community-map public
context mycontext
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show running-config ssh

Displays the Secure Shell (SSH) status in the running-config.

## Syntax

```
show running-config ssh
```

## Modes

Privileged EXEC mode

# show running-config ssh server

Displays the SSH server status in the running-config.

## Syntax

```
show running-config [ rbridge-id { rbridge-id | all } ] ssh server
```

## Parameters

- rbridge-id**  
Specifies an RBridge or all RBridges.
- rbridge-id*  
Specifies an RBridge ID.
- all**  
Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

SSH server configuration is placed at the beginning of the running-config and is part of the global configuration of the device. SSH is enabled by default and no entry is shown in the running-config when set to default.

## Examples

When SSH service is shut down:

```
device# show running-config rbridge-id 3 ssh server
rbridge-id 3
ssh server shutdown
device# show running-config rbridge-id 3 ssh server
rbridge-id 3
ssh server shutdown
ssh server key-exchange dh-group-14
```

When SSH service is enabled:

```
device# show running-config rbridge-id
rbridge-id
| all] ssh server
% No entries found
```

## Related Commands

[show ssh server status](#), [ssh server shutdown](#)

# show running-config ssh server key-exchange

Displays the SSH server key-exchange status in the running-config.

## Syntax

```
show running-config ssh server key-exchange
```

## Modes

Privileged EXEC mode

## Examples

Typical command output:

```
device# show running-config ssh server key-exchange
ssh server key-exchange dh-group-14
```

When SSH Server Key-exchange is configured to DH Group 14:

```
device# show running-config rbridge-id 3 ssh server key-exchange
rbridge-id 3
ssh server key-exchange dh-group-14
```

When SSH Server Key-exchange method has the default value:

```
device# show running-config rbridge-id 3 ssh server key-exchange
rbridge-id 3
```

## Related Commands

[show ssh server status](#)

# show running-config support autoupload-param

Displays autoupload parameters.

## Syntax

```
show running-config support autoupload-param
```

## Modes

Privileged EXEC mode

## Examples

```
switch(config)# do show running-config support autoupload-param
support autoupload-param hostip 10.31.2.27 username hegdes directory /users/home40/hegdes/autoupload
protocol ftp password "3iTYxJWEUHp9axZQt2tbvw==\n"
switch(config)#
```

# show running-config support support-param

Displays support parameters

## Syntax

```
show running-config support support-param [ hostip host-ip user user_acct password password protocol [ ftp | scp | sftp ]
    directory path ]
```

## Parameters

**hostip** *host-ip*

Displays the IP address of the remote host.

**user** *user\_acct*

Displays the user name to access the remote host.

**password** *password*

Displays the password to access the remote host.

**protocol** FTP | SCP | SFTP

Displays the protocol used to access the remote server.

**directory** *path*

Displays the path to the directory.

## Modes

Privileged EXEC mode

## Examples

```
switch(config)# do show running-config support support-param
support support-param hostip 10.31.2.27 username hegdes directory /users/home40/hegdes/support protocol
ftp password "3iTYxJWEUHp9axZQt2tbvw==\n"
switch(config)#
```

## History

Release version	Command history
5.0.0	This command was introduced.

# show running-config switch-attributes

Displays switch attributes.

## Syntax

```
show running-config switch-attributes [ rbridge-id ] { chassis-name | host-name }
```

## Command Default

Displays all switch attributes on the local switch. The default host name is "sw0". The default chassis name depends on the switch model.

## Parameters

*rbridge-id*

Specifies an RBridge ID.

**chassis-name**

Displays the switch chassis name.

**host-name**

Displays the switch host name.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

This command is not supported on the standby management module.

## Examples

To display all attributes for the local switch:

```
switch# show running-config switch-attributes

switch-attributes 2
  chassis-name VDX6740-48
  host-name sw0
!
```

To display the host name of the local switch:

```
switch# show running-config switch-attributes host-name

switch-attributes 2
  host-name sw0
!
```

## Related Commands

[switch-attributes](#)

# show running-config system-monitor

Displays the system monitor configuration.

## Syntax

```
show running-config system-monitor [ fan | power | temp | cid-card | sfp | compact-flash | MM | LineCard | SFM ]
```

## Parameters

### fan

Displays the threshold and alert setting for the FAN component.

### power

Displays the threshold and alert setting for the power component.

### temp

Displays the threshold for the temperature sensor component.

### cid-card

Displays the threshold for the CID card component.

### sfp

Displays the threshold for the small form factor pluggable (SFP) device.

### compact-flash

Displays the threshold for the compact flash device.

### MM

Displays the threshold for the management module.

### LineCard

Displays the threshold for the line card.

### SFM

Displays the threshold for the switch fabric module.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.



## Examples

```
switch# show running-config system-monitor

system-monitor fan threshold marginal-threshold 1 down-threshold 2
system-monitor fan alert state removed action raslog
system-monitor power threshold marginal-threshold 0 down-threshold 1
system-monitor power alert state removed action raslog
system-monitor temp threshold marginal-threshold 1 down-threshold 2
system-monitor cid-card threshold marginal-threshold 1 down-threshold 0
system-monitor cid-card alert state inserted,faulty action email
system-monitor sfp alert state none action none
system-monitor compact-flash threshold marginal-threshold 1 down-threshold 0
system-monitor MM threshold marginal-threshold 1 down-threshold 0
system-monitor LineCard threshold marginal-threshold 1 down-threshold 0
system-monitor LineCard alert state removed action raslog
system-monitor SFM threshold marginal-threshold 1 down-threshold 0
```

## Related Commands

[show system monitor](#), [system-monitor-mail](#)

# show running-config system-monitor-mail

Displays the system monitor mail configuration.

## Syntax

```
show running-config system-monitor-mail { fru enable }
```

## Parameters

**fru**

Displays FRU information.

**enable**

Displays the status of the FRU.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

## Related Commands

[show system monitor](#), [system-monitor-mail](#)

# show running-config tacacs-server

Displays the TACACS+ server configuration.

## Syntax

```
show running-config tacacs-server [ host ipaddr | hostname ]
```

## Parameters

### host

Identifies the TACACS+ server by host name or IP address.

### *ipaddr*

Specifies the IP address of the TACACS+ server (IPv4 or IPv6).

### *hostname*

Specifies the domain name of the TACACS+ server.

### source-ip [*chassis-ip* | *mm-ip*]

Specifies the chassis IP address or MM IP address as the source IP address for TACACS+ authentication and accounting.

## Modes

Privileged EXEC mode

## Examples

To display the list of configured TACACS+ servers:

```
switch# show running-config tacacs-server host  
fec0:60:69bc:94:211:25ff:fec4:6010
```

To display a single IPv4 TACACS+ server configuration:

```
switch# show running-config tacacs-server host 10.24.65.6
```

To display a single IPv6 TACACS+ server configuration:

```
switch# show running-config tacacs-server host fec0:60:69bc:94:211:25ff:fec4:6010
```

## Related Commands

[radius-server](#), [show running-config radius-server](#), [tacacs-server](#)

# show running-config telnet server

Displays the Telnet server status in the running-config.

## Syntax

```
show running-config [ rbridge-id { rbridge-id | all } ] telnet server
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Telnet server configuration is placed at the beginning of running-config and is part of the global configuration of the switch. Telnet is enabled by default and there will be no entry in the running-config when set to default.

## Examples

When Telnet service is shut down:

```
switch# show running-config rbridge-id 3 telnet server
rbridge-id 3
telnet server shutdown
```

When Telnet service is enabled:

```
switch# show running-config [rbridge-id
rbridge-id
| all] telnet server
% No entries found
```

## Related Commands

[show telnet server status](#), [telnet](#), [telnet server shutdown](#)

# show running-config threshold-monitor

Displays the system's threshold configuration.

## Syntax

```
show running-config threshold-monitor
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

## Examples

```
switch# show running-config threshold-monitor

threshold-monitor Cpu poll 30 retry 2 limit 60 actions raslog
threshold-monitor Memory poll 30 retry 2 limit 70 high-limit low-limit 50 actions none
switch# show running-config threshold-monitor area IFG
```

Interface	Area	Value	Status	Monitoring Status
fortygigabitethernet 3/8	IFG Violation Error	30	Out of Range	Monitoring
fortygigabitethernet 3/9	IFG Violation Error	0	In Range	Monitoring
<All other online interfaces>	IFG Violation Error	0	In Range	Monitoring

## Related Commands

[show system monitor](#), [system-monitor-mail](#)

# show running-config threshold-monitor interface

Displays the system's running interface configuration.

## Syntax

**show running-config threshold-monitor interface**

## Modes

Privileged EXEC mode

## Usage Guidelines

Default values are not displayed under the **show running-config threshold-monitor interface** command. Only custom values are displayed.

## Examples

```
switch# do show running-config threshold-monitor interface

switch(config)# do show running-config threshold-monitor interface
threshold-monitor interface apply custom-monitoring
threshold-monitor interface pause
threshold-monitor interface policy custom type Ethernet area MissingTerminationCharacter threshold
timebase minute high-threshold 20 low-threshold 1 buffer 5
threshold-monitor interface policy custom type Ethernet area MissingTerminationCharacter alert above
highthresh-action none lowthresh-action none
threshold-monitor interface policy custom type Ethernet area MissingTerminationCharacter alert below
highthresh-action none lowthresh-action none
threshold-monitor interface policy custom type Ethernet area CRCAlignErrors threshold timebase hour
high-threshold 80 low-threshold 10 buffer 35
threshold-monitor interfacepolicy custom type Ethernet area CRCAlignErrors alert above highthresh-
action none lowthresh-action none
threshold-monitor interface policy custom type Ethernet area CRCAlignErrors alert below highthresh-
action none lowthresh-action none
threshold-monitor interface policy custom type Ethernet area SymbolErrors threshold timebase minute
high-threshold 20 low-threshold 1 buffer 5
threshold-monitor interfacepolicy custom type Ethernet area SymbolErrors alert above highthresh-action
none lowthresh-action none
threshold-monitor interface policy custom type Ethernet area SymbolErrors alert below highthresh-action
none lowthresh-action none
threshold-monitor interface policy custom type Ethernet area IFG threshold timebase minute high-
threshold 20 low-threshold 1 buffer 5
threshold-monitor interface policy custom type Ethernet area IFG alert above highthresh-action
raslog,portfence lowthresh-action emailraslog
threshold-monitor interface policy custom type Ethernet area IFG alert below highthresh-action none
lowthresh-action none
```

## Related Commands

[show system monitor](#)

# show running-config threshold-monitor security

Displays the system's running security configuration.

## Syntax

```
show running-config threshold-monitor security
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Default values are not displayed under the **show running-config threshold-monitor security** command. Only custom values are displayed.

## Examples

```
switch# show running-config threshold-monitor security policy custom area telnet-violation
threshold-monitor security policy custom area telnet-violation timebase hour
threshold-monitor security policy custom area telnet-violation threshold thresh_high high-threshold 10
buffer 20
switch# show running-config threshold-monitor policy custom area login-violation
threshold-monitor securitym policy custom area login-violation alert above highthresh_action all
threshold-monitor security apply custom
switch#
```

## Related Commands

[show system monitor](#)

# show running-config threshold-monitor sfp

Displays the system's running SFP configuration.

## Syntax

```
show running-config threshold-monitor sfp
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Default values are not displayed under the **show running-config threshold-monitor sfp** command. Only custom values are displayed.

## Examples

```
switch# do show running-config threshold-monitor sfp

threshold-monitor sfp pause
threshold-monitor sfp apply custom
threshold-monitor sfp policy custom Type 1GSR area TXP threshold high-threshold 2000 low-threshold 1000
buffer 500
threshold-monitor sfp policy custom Type 1GSR area TXP alert above highthresh-action raslog lowthresh-
action none
threshold-monitor sfp policy custom Type 1GSR area TXP alert below highthresh-action none lowthresh-
action raslog
```

## Related Commands

[show system monitor](#)



# show running-config username

Displays the user accounts on the device.

## Syntax

```
show running-config username [ username ] [ access-time ] [ desc ] [ enable ] [ encryption-level ] [ expire ] [ password ] [ role ]
```

## Parameters

### *username*

Displays the configuration of a specified username. The maximum number of characters is 40.

### **access-time**

Displays access-time configuration.

### **desc**

Displays the description of the user configuration.

### **enable**

Displays the account enablement status.

### **encryption-level**

Password encryption level. Values are 0 through 7. The default is 0.

### **expire**

Date until the password remains valid in YYYY-MM-DD format. Valid year values range from 1902 through 2037. By default, passwords do not expire.

### **password**

Account password.

### **role**

The role associated with the account.

## Modes

Privileged EXEC mode

## Usage Guidelines

To display details for one user only, specify *username* . Otherwise, this command displays all user accounts on the device.

Use the various parameters to query the specified account details.

This command does not display the root account.

Defaults are not displayed.

## Examples

The following example displays the user accounts on the device.

```
device# show running-config username

username admin password "BwrsDbB+tABWGWpINOVKoQ==\n" encryption-level 7 role admin desc Administrator
username user password "BwrsDbB+tABWGWpINOVKoQ==\n" encryption-level 7 role user desc User
```

The following example displays a specific user account.

```
device# show running-config username admin

username admin password "BwrsDbB+tABWGWpINOVKoQ==\n" encryption-level 7 role admin desc Administrator
```

The following example displays the enabled status for a specific user account.

```
device# show running-config username admin enable

username admin enable true
```

The following example displays user access on the device.

```
device# show running-config username access-time
username admin access-time ""
username extremel access-time 0000
username user access-time ""
username user1 access-time 1700
```

## Related Commands

[show users](#), [unlock username](#), [username](#)

# show running-config vcs

Displays VCS configuration information.

## Syntax

```
show running-config vcs [ virtual [ ip [ address ] ] ]
```

## Parameters

### virtual

Displays the VCS configuration.

### ip

Displays the virtual IP configuration.

### address

Displays the virtual IP address.

## Modes

Privileged EXEC mode

## Related Commands

[vcs config snapshot](#)

# show running-config vlag-commit-mode

Displays the state of the virtual LAG (vLAG) commit mode for dynamic vLAGs for vLAG scalability.

## Syntax

`show running-config vlag-commit-mode`

## Modes

Privileged EXEC mode

## Usage Guidelines

`vlag-commit-mode disable`

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[vlag-commit-mode disable](#)

# show running-config zoning

Displays the zoning configuration.

## Syntax

```
show running-config zoning [ defined-configuration | enabled-configuration ]
```

## Parameters

### defined-configuration

Displays the defined configuration parameters. See **show running-config zoning defined-configuration**.

### enabled-configuration

Displays the enabled configuration parameters. See **show running-config zoning enabled-configuration**.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the zoning configuration for an Extreme VCS Fabric. The base command lists both the defined and the enabled configuration.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

To display the enabled-configuration enabled-zone information, use the **show zoning enabled-configuration** command.

This command can be entered on any R Bridge in an Extreme VCS Fabric.

## Examples

The following example displays the zoning configuration.

```
device# show running-config zoning
zoning defined-configuration cfg cfg1
  member-zone zone1
  member-zone zone2
```

## Related Commands

[show running-config zoning defined-configuration](#), [show running-config zoning enabled-configuration](#), [show zoning enabled-configuration](#)

# show running-config zoning defined-configuration

Displays the defined zone configuration.

## Syntax

`show running-config zoning defined-configuration`

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the zoning configuration for an Extreme VCS Fabric. The base command lists both the defined and a small subset of the enabled configuration.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric.

## Examples

The following example displays the defined zoning configuration:

```
device# show running-config zoning defined-configuration
```

```
zoning defined-configuration cfg cfg0
member-zone zone_0_1
member-zone zone_0_2
member-zone zone_0_3
member-zone zone_0_4
member-zone zone_same
!
zoning defined-configuration cfg cfg1
member-zone zone_1_1
member-zone zone_1_2
member-zone zone_1_3
member-zone zone_1_4
member-zone zone_same
!
zoning defined-configuration cfg cfg2
member-zone zone_2_1
member-zone zone_2_2
member-zone zone_2_3
member-zone zone_2_4
member-zone zone_same
!
zoning defined-configuration cfg cfg4
member-zone zone2
member-zone zone3
!
zoning defined-configuration zone zone0
member-entry 11:22:33:44:55:66:77:80
member-entry 11:22:33:44:55:66:77:81
member-entry 11:22:33:44:55:66:77:82
member-entry 11:22:33:44:55:66:77:83
member-entry 11:22:33:44:55:66:77:84
!
zoning defined-configuration zone zone1
member-entry 11:22:33:44:55:66:77:80
member-entry 11:22:33:44:55:66:77:81
member-entry alias1
member-entry alias2
member-entry alias3
!
zoning defined-configuration zone zone2
member-entry 11:22:33:44:55:66:77:80
member-entry 11:22:33:44:55:66:77:81
member-entry 11:22:33:44:55:66:77:82
member-entry 11:22:33:44:55:66:77:83
member-entry 11:22:33:44:55:66:77:84
!
```

## Related Commands

[show running-config zoning](#), [show running-config zoning enabled-configuration](#)

# show running-config zoning enabled-configuration

Displays the enabled zone configuration.

## Syntax

```
show running-config zoning enabled-configuration [ cfg-action | cfg-name | default-zone-access ]
```

## Parameters

### cfg-action

Displays the enabled configuration action.

### cfg-name

Displays the enabled configuration name.

### default-zone-access

Displays the default-zone access mode.

## Modes

Privileged EXEC mode

## Usage Guidelines

The enabled configuration is the single zone configuration that is currently enabled in an Extreme VCS Fabric. The devices an initiator can access in the network are based on this configuration. The enabled configuration is built when a specific zone configuration is enabled and all error checking has been completed successfully. Use this command to display the name of the enabled zoning configuration, the configuration action, and the default-zone access mode.

The configuration name is displayed differently depending on two main factors:

- If no transaction is pending, the configuration action is set to "cfg-save."
- If a transaction is pending, the configuration name is marked with a CFG\_MARKER asterisk (\*) and the configuration action is set to "cfg-None" The CFG\_MARKER flag can also indicate that the enabled configuration does not exactly match the defined configuration. This scenario occurs when you have an enabled configuration and make edits to the defined configuration, and then, instead of enabling the defined configuration, you issue the cfg-save command.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be executed on any R Bridge in an Extreme VCS Fabric.

To view details about the enabled zones, use the **show zoning enabled-configuration** command.

## Examples

To display the enabled zoning configuration:

```
device# show running-config zoning enabled-configuration
zoning enabled-configuration cfg-name cfg1
zoning enabled-configuration default-zone-access noaccess
zoning enabled-configuration cfg-action cfg-save
```



To display only the default-zone access mode for the enabled zoning configuration:

```
device# show running-config zoning enabled-configuration default-zone-access
zoning enabled-configuration default-zone-access allaccess
```

## Related Commands

[show running-config zoning](#), [show running-config zoning defined-configuration](#), [show zoning enabled-configuration](#)

# show secpolicy

Displays the Switch Connection Control (SCC) security policy information.

## Syntax

```
show running-config secpolicy { defined-policy | active-policy }
```

## Parameters

### defined-policy

Displays the defined policy and its policy member set.

### active-policy

Displays the active policy and its policy member set.

## Modes

Privileged EXEC mode

## Examples

To show only the defined policy

```
switch# show running-config secpolicy defined-policy

secpolicy defined-policy SCC_POLICY
member-entry 11:11:11:11:11:11:11:11
!
member-entry 22:22:22:22:22:22:22:22
!
member-entry 33:33:33:33:33:33:33:33
```

To show only the active policy

```
switch# show running-config secpolicy active-policy

secpolicy active-policy SCC_POLICY
member-entry 11:11:11:11:11:11:11:11
!
member-entry 22:22:22:22:22:22:22:22
!
member-entry 33:33:33:33:33:33:33:33
```

To show both active and defined policy

```
switch# show running-config secpolicy

secpolicy defined-policy SCC_POLICY
member-entry 11:11:11:11:11:11:11:11
!
member-entry 22:22:22:22:22:22:22:22
!
member-entry 33:33:33:33:33:33:33:33
!
!
secpolicy active-policy SCC_POLICY
member-entry 11:11:11:11:11:11:11:11
!
member-entry 22:22:22:22:22:22:22:22
!
member-entry 33:33:33:33:33:33:33:33
```

## Related Commands

[secpolicy activate](#), [secpolicy defined-policy](#)

# show sflow

Displays sFlow configuration information and statistics.

## Syntax

```
show sflow [ interface { <N>gigabitethernet rbridge-id/slot/port | all
```

## Command Default

sFlow is disabled on all interfaces.

## Parameters

### all

Displays all sFlow information and statistics.

### interface

Displays sFlow information for an Ethernet interface.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Examples

To display sFlow statistics and view the configured VRFs:

```
device# show sflow
sFlow services are:                disabled
Global default sampling rate:      32768 pkts
Global default counter polling interval: 20 secs
Rbridge-Id          Collector server address          vrf-name          Samples sent
-----
-
      161                10.1.1.1:6343                default-vrf                0
      161                10.1.1.2:6343                mgmt-vrf                  0
```

To display sFlow statistics on the 10-gigabit Ethernet interface 15/0/4:

```
device# show sflow interface tengigabitethernet 15/0/4

sFlow info for interface Ten Gigabit Ethernet 15/0/4
-----
Configured sampling rate:          100 pkts
Actual sampling rate:             100 pkts
Counter polling interval:         100 secs
Samples received from hardware:    32
Port backoffThreshold :           272
Counter samples collected :       147
```

To display sFlow statistics on 1-gigabit Ethernet interface 22/0/1:

```
device# show sflow interface gigabitethernet 22/0/1

-----
sFlow info for interface Gigabit Ethernet 22/0/1
Configured sampling rate:          32768 pkts
Actual sampling rate:             32768 pkts
Counter polling interval:         20 seconds
Samples received from hardware:    0
Port backoff threshold:           48
-----
```

To display all sFlow statistics:

```
device# show sflow all
sFlow services are:                               enabled
Global default sampling rate:                    32768 pkts
Global default counter polling interval:         20 secs
Collector server address                          Number of samples sent
-----
3ffe:1900:4545:3:200:f8ff:fe21:67cf : 6343      0
fe80::200:f8ff:fe21:67cf : 6343              0
192.35.41.32 : 6343                          0
fe80::201:fdff:fe21:43cd : 6343              0
192.44.23.45 : 6343                          0
```

# show sflow-profile

Displays the sflow profile configurations.

## Syntax

```
show sflow-profile { string | all }
```

## Parameters

*string*

Specifies the name of the profile.

**all**

Displays all profile information.

## Modes

Privileged EXEC mode

## Related Commands

[sflow enable \(global version\)](#)

# show sfm

Displays information about the switch fabric modules present in the chassis.

## Syntax

```
show sfm [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID. The range of valid values is from 1 through 239.

## Modes

Privileged EXEC mode

## Command Output

The **show sfm** command displays the following information:

Output field	Description
Slot	Displays the slot number. Slots for device fabric modules are S1 through S3 for Extreme VDX 8770-4 devices and S1 through S6 for Extreme VDX 8770-8 devices.
Type	Displays the interface module type. The switch fabric module type is SFM.
Description	Module description
ID	Displays the module ID. The ID for the switch fabric module is 113.
Status	Displays the status of the module as one of the following: <ul style="list-style-type: none"> <li>VACANT - The slot is empty.</li> <li>POWERED-OFF - The module is present in the slot but is powered off.</li> <li>POWERING UP - The module is present and powering on.</li> <li>LOADING - The module is present, powered on, and loading the initial configuration.</li> <li>DIAG RUNNING POST1 - The module is present, powered on, and running the POST (power-on self-test).</li> <li>DIAG RUNNING POST2 - The module is present, powered on, and running the reboot power on self tests.</li> <li>INITIALIZING - The module is present, powered on, and initializing hardware components.</li> <li>ENABLED - The module is on and fully enabled.</li> <li>DISABLED - The module is powered on but disabled.</li> <li>FAULTY - The module is faulty because an error was detected.</li> <li>UNKNOWN -The module is inserted but its state cannot be determined.</li> </ul>

## Examples

The following example displays the switch fabric modules in an Extreme VDX 8770-4 chassis.

```
device# show sfm
```

```
Slot  Type          Description          ID      Status
-----
S1   SFM              Switch Fabric Module  113    ENABLED#
S2   SFM              Switch Fabric Module  113    ENABLED#
S3   SFM              Switch Fabric Module  113    ENABLED
# = At least one enabled SFM in these slots is required.
```

The following example displays the switch fabric modules on RBridge 80.

```
device# show sfm rbridge-id 80
Rbridge-id 80:
```

```
Slot  Type          Description          ID      Status
-----
S1   SFM              Switch Fabric Module  113    ENABLED#
S2   SFM              Switch Fabric Module  113    ENABLED#
S3                   VACANT
# = At least one enabled SFM in these slots is required.
```

## History

Release version	Command history
6.0.0	This command was modified to support the <b>rbridge-id</b> parameter.

## Related Commands

[show linecard](#), [show mm](#), [show slots](#)



# show sfp

Displays the SFP breakout configurations.

## Syntax

```
show sfp [ linecard linecard [ port port ] ]
```

## Command Default

Displays the SFP breakout information using a line card. Port number is optional. If absent, all SFP port configurations are shown.

## Parameters

**linecard** *linecard*

Specifies line card information.

**port** *port*

Specifies port information.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the SFP configuration for the specified line cards. The speed column shows the breakout speed in breakout mode, and the aggregate speed when not in breakout mode.

This command is supported on the line cards.

## Examples

To display the SFP configuration on a line card:

```
switch# show sfp linecard 1
Port      Type      Breakout      Speed
-----
1         SFP       n/a           10G
2         QSFP      4x10G         10G
3         SFP       n/a           40G
4         CSFP      10x10G        100G
switch# show sfp linecard 1 port 2
Port      Type      Breakout      Speed
-----
2         QSFP      4X10G         10G
```

# show slots

Displays information about the modules present in the chassis.

## Syntax

```
show slots [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID. The range of valid values is from 1 through 239.

## Command Output

The **show slots** command displays the following information:

Output field	Description
Slot	Displays the slot number. Slots for interface modules are L1 through L4 on Extreme VDX 8770-4 devices, and L1 through L8 on the Extreme VDX 8770-8 devices.
Type	Displays the module type. Examples are: <ul style="list-style-type: none"> <li>• MM Management Module</li> <li>• SFM Switch Fabric Module</li> <li>• LC48X10G 48-port 10 GbE interface module (line card)</li> <li>• LC48X1G 48-port 1 GbE interface module</li> <li>• LC12X40G 12-port 40 GbE interface module</li> <li>• 27x40G 27-port 40 GbE interface module</li> <li>• 6x100G 6-port 100 GbE interface module</li> </ul>
Description	Module description
ID	Module ID. Examples are: <ul style="list-style-type: none"> <li>• 112 Management Module</li> <li>• 113 Switch Fabric Module</li> <li>• 114 48-port 10GbE interface module</li> <li>• 127 12-port 10 GbE interface module</li> </ul>
Status	Displays the status of the module as one of the following: <ul style="list-style-type: none"> <li>• VACANT - The slot is empty.</li> <li>• POWERED-OFF - The module is present in the slot but is powered off.</li> <li>• POWERING UP - The module is present and powering on.</li> <li>• LOADING - The module is present, powered on, and loading the initial configuration.</li> <li>• DIAG RUNNING POST1 - The module is present, powered on, and running the POST (power-on self-test). This status is not valid for the management modules.</li> <li>• DIAG RUNNING POST2 - The module is present, powered on, and running the reboot power on self tests. This status is not valid for the management modules.</li> <li>• INITIALIZING - The module is present, powered on, and initializing hardware components.</li> <li>• ENABLED - The module is on and fully enabled.</li> <li>• DISABLED - The module is powered on but disabled.</li> <li>• FAULTY - The module is faulty because an error was detected.</li> </ul>

Output field	Description
	<ul style="list-style-type: none"> <li>UNKNOWN - The module is inserted but its state cannot be determined.</li> </ul>

## Modes

Privileged EXEC mode

## Examples

### NOTE

An "@" following an SFM status line indicates that the status of the optical switch is "OPEN."

The following example displays the modules in an Extreme VDX 8770-4 chassis.

```
device# show slots
```

```
Slot  Type          Description                      ID      Status
-----
M1    MM                Management Module                112     ENABLED
M2    MM                Management Module                112     VACANT
S1    SFM               Switch Fabric Module             113     VACANT#
S2    SFM               Switch Fabric Module             113     ENABLED#
S3    SFM               Switch Fabric Module             113     VACANT
L1    LC48X10G          48-port 10GE card                114     ENABLED
L2    LC48X10G          48-port 10GE card                114     ENABLED
L3    LC48X10G          48-port 10GE card                114     VACANT
L4    LC48X1G           48-port 1GE card                 131     ENABLED
# = At least one enabled SFM in these slots is required.
```

The following example displays the modules in RBridge 80.

```
device# show slots rbridge-id 80
Rbridge-id 80:
```

```
Slot  Type          Description                      ID      Status
-----
M1    MM                Management Module                112     ENABLED
M2    MM                Management Module                112     ENABLED
S1    SFM               Switch Fabric Module             113     ENABLED#
S2    SFM               Switch Fabric Module             113     ENABLED#
S3    SFM               Switch Fabric Module             113     VACANT
L1    LC6X100G          6-port 100GE card                149     ENABLED
L2    LC48X10G          48-port 10GE card                114     ENABLED
L3    LC48X10G          48-port 10GE card                114     ENABLED
L4    LC12X40G          12-port 40GE card                127     ENABLED
# = At least one enabled SFM in these slots is required.
```

## History

Release version	Command history
6.0.0	This command was modified to the support the <b>rbridge-id</b> parameter.

## Related Commands

[show linecard](#), [show sfm](#)

show span path

# show span path

Displays the SPAN path information.

## Syntax

```
show span path session session_number
```

## Parameters

**session** *session\_number*

The path for the SPAN session to display.

## Modes

Privileged EXEC

## Examples

The following example displays the SPAN path information.

```
device# show span path session 1

Session                :1
Path                   :Te 1/0/10 -> Te 1/0/1 (ISL-exit port) -> Te 2/0/16
```

## Related Commands

[monitor session, source](#)

# show spanning-tree

Displays Spanning Tree Protocol (STP) information.

## Syntax

```
show spanning-tree [ brief | interface { ethernet slot/port | port-channel port_channel_number } | pvst | mst [ brief | detail |
instance instance_id | interface ] mst-config | vlan vlan_id ]
```

## Parameters

### brief

Display brief spanning tree information.

### interface

Display information about the spanning tree configuration on an interface.

### ethernet *slot/port*

Display spanning tree information about a specific Ethernet interface.

### port-channel *port\_channel\_number*

Display spanning tree information about a port channel interface.

### pvst

Display PVST+ information.

### mst

Display MSTP information.

### detail

Display detailed MSTP tree information.

### instance *instance\_id*

Display MSTP information about a specific instance.

### mst-config

Display MSTP region configuration information.

### vlan *vlan\_id*

Display spanning tree information about a specific VLAN.

## Modes

Privileged EXEC mode.

## Usage Guidelines

On the VDX family of switches, VLANs are treated as interfaces from a configuration point of view. By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). Valid VLAN IDs are as follows:

- On VDX 8770 switches: 1 through 4086 for 802.1Q VLANs (VLAN IDs 4087 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.

show spanning-tree

- On all other VDX switches: 1 through 3962 for 802.1Q VLANs (VLAN IDs 3963 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.

#### NOTE

Extreme Networks supports the PVST+ and R-PVST+ protocols. The PVST and R-PVST protocols are proprietary to Cisco and are not supported.

## Examples

To display spanning tree information:

```
device# show spanning-tree brief
```

```
Spanning-tree Mode: Spanning Tree Protocol
```

```
Root ID      Priority 4096
             Address 768e.f805.5800
             Hello Time 8, Max Age 25, Forward Delay 20
```

```
Bridge ID    Priority 4096
             Address 768e.f805.5800
             Hello Time 8, Max Age 25, Forward Delay 20
```

Interface	Role	Sts	Cost	Prio	Link-type	Edge
Te 1/0/11	DES	FWD	2000	128	P2P	No
Te 1/0/12	DES	FWD	2000	128	P2P	No
Te 1/0/14	ALT	DSC	2000	128	P2P	No

Typical output of a summary that contains an Rbridge-id as a non-root port.

```
switch# show spanning-tree brief
```

```
Spanning-tree Mode: Rapid Spanning Tree Protocol
```

```
Root ID      Priority 4096
             Address 0005.1ecd.0b8a
             Hello Time 8, Max Age 25, Forward Delay 20
```

```
Root Port ID : 5/0/22
Bridge ID    Priority 4096
             Address 0105.3352.6f27
             STP Switch Id: 01e0.5200.0211
             Hello Time 8, Max Age 25, Forward Delay 20
             Migrate Time 3 sec
```

Interface	Role	Sts	Cost	Prio	Link-type	Edge
Te 6/0/20	DES	FWD	2000	128	P2P	No
Te 6/0/21	DES	FWD	2000	128	P2P	No
Te 6/0/23	ALT	DSC	2000	128	P2P	No

## Related Commands

[show spanning-tree interface](#)

# show spanning-tree brief

Displays the status and parameters of the Spanning Tree Protocol (STP).

## Syntax

```
show spanning-tree [ vlan vlan_id ] brief
```

## Parameters

**vlan *vlan\_id***  
Specifies a VLAN.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display a summary of the status and parameters of STP for each interface, including the port roles and port states.

The following describes the port roles and states:

- Port roles—root port, designated port, alternate port and backup port.
- Port states—discarding, learning, forwarding, and blocked.
- Port types—edge port (PortFast), point-to-point, and shared port.

When "root guard" is in effect, the **show spanning-tree brief** command output shows the port state as ERR, not root\_inc.

## Examples

To display the interface summary of the Spanning Tree Protocol:

```
switch# show spanning-tree brief

Spanning-tree Mode: Rapid Spanning Tree Protocol
  Root ID          Priority 32768
                  Address 0005.1e76.1aa0
                  Hello Time 2, Max Age 20, Forward Delay 15
  Bridge ID       Priority 32768
                  Address 0005.1e76.1aa0
                  Hello Time 2, Max Age 20, Forward Delay 15, Tx-HoldCount 6
                  Migrate Time 3 sec

Interface    Role  Sts  Cost      Prio  Link-type    Boundary  Edge
-----
Te 0/0       DIS  DSC  2000      128   P2P          Yes       No
Te 0/1       ALT  BLK  2000      128   P2P          Yes       No
Te 0/2       RTPT BLK  2000      128   P2P          Yes       No
Te 0/3       DIS  BLK  2000      128   P2P          Yes       No
Te 0/8       DIS  DSC  2000      128   P2P          Yes       No
Te 0/19      DIS  DSC  2000      128   P2P          Yes       No
Te 0/20      DIS  DSC  2000      128   P2P          Yes       No
```

show spanning-tree brief

Typical output of a summary that contains an rbridge-id as a non-root port.

```
switch# show spanning-tree brief
Spanning-tree Mode: Rapid Spanning Tree Protocol
  Root ID          Priority 32768
                Address 0005.1ecd.0b8a
                Hello Time 2, Max Age 20, Forward Delay 15
  Root Port ID : 5/0/22
  Bridge ID       Priority 32768
                Address 0105.3352.6f27
                STP Switch Id: 01e0.5200.0211
                Hello Time 2, Max Age 20,
                Forward Delay 15, Tx-HoldCount 6
                Migrate Time 3 sec
Interface      Role  Sts  Cost      Prio  Link-type      Edge
-----
Te 6/0/20      DES  FWD  2000      128   P2P             No
Te 6/0/21      DES  FWD  2000      128   P2P             No
Te 6/0/23      ALT  DSC  2000      128   P2P             No
```

## Related Commands

[show spanning-tree interface](#)



# show spanning-tree interface

Displays the state of the Spanning Tree Protocol for all named port-channels or 1-gigabit Ethernet, or 10-gigabit Ethernet interfaces.

## Syntax

```
show spanning-tree interface [ port-channel number | <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

**port-channel** *number*

Specifies the port-channel number. The number of available channels range from 1 through 6144.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

The following describes the port roles, states and types:

- Port roles—root port, designated port, alternate port and backup port.
- Port states—discarding, learning, and forwarding.
- Port types—edge port (PortFast), point-to-point, and shared port.

show spanning-tree interface

To display information on a port channel:

```
switch# show spanning-tree interface port-channel 101

Spanning-tree Mode: Multiple Spanning Tree Protocol

Root Id: 8000.0023.04ee.be01
Bridge Id: 8000.01e0.5200.c970

Port Po 101 enabled
vLAG STP Master Rbridge Id: 2
Ifindex: 671088741; Id: 8007; Role: Rootport; State: Forwarding
Designated External Path Cost: 0; Internal Path Cost: 0
Configured Path Cost: 2000
Designated Port Id: 9064; Port Priority: 128
Designated Bridge: 8000.0023.04ee.be01
Number of forward-transitions: 7
Version: Multiple Spanning Tree Protocol - Received MSTP - Sent MSTP
Edgeport: off; AutoEdge: no; AdminEdge: no; EdgeDelay: 3 sec
Restricted-role is disabled
Restricted-tcn is disabled
Boundary: no
Bpdu-guard: off
Bpdu-filter: off
Link-type: point-to-point
Peer-switch: enabled
Received BPDUs: 9914; Sent BPDUs: 42
```

## Related Commands

[show spanning-tree brief](#)

# show spanning-tree mst brief

Displays the status and parameters of the Multiple Spanning Tree Protocol (MSTP) instance information in brief.

## Syntax

```
show spanning-tree mst brief
```

## Modes

Privileged EXEC mode

## Usage Guidelines

The command output includes the port roles, port states and port types.

- Port roles—root port, designated port, alternate port, and backup port.
- Port states—discarding, learning, and forwarding.
- Port types—edge port (PortFast), point-to-point, and shared port.

## Examples

To display the status and parameters of the MSTP instance information:

```
switch# show spanning-tree mst brief

Spanning-tree Mode: Multiple Spanning Tree Protocol
CIST Root ID          Priority 32768
                     Address 0005.1e76.1aa0
CIST Bridge ID        Priority 32768
                     Address 0005.1e76.1aa0
CIST Regional Root ID Priority 32768
                     Address 0005.1e76.1aa0
Configured Hello Time 2, Max Age 20, Forward Delay 15
Max Hops 20, Tx-HoldCount 6
CIST Root Hello Time 2, Max Age 20, Forward Delay 15, Max Hops 20
CIST Root path cost 0
Interface   Role  Sts  Cost      Prio  Link-type      Boundary  Edge
-----
Te 0/0      DIS  DSC  2000      128   P2P            Yes       No
Te 0/1      ALT  BLK  2000      128   P2P            Yes       No
Te 0/2      RTPT BLK  2000      128   P2P            Yes       No
Te 0/3      DIS  BLK  2000      128   P2P            Yes       No
Te 0/8      DIS  DSC  2000      128   P2P            Yes       No
Te 0/19     DIS  DSC  2000      128   P2P            Yes       No
Te 0/20     DIS  DSC  2000      128   P2P            Yes       No
```

## Related Commands

[show spanning-tree mst instance](#), [show spanning-tree mst interface](#)

# show spanning-tree mst detail

Displays details on an interface for the Multiple Spanning Tree Protocol (MSTP) instance running.

## Syntax

```
show spanning-tree mst detail [ interface port-channel number | interface <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### interface

Specifies the interface for which to display the MSTP information.

### port-channel *number*

Specifies the port-channel of the interface. The number of available channels range from 1 through 6144.

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **ten**gigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **gigabitethernet** *rbridge-id/slot/port* parameter is used only on Extreme VDX 6710, Extreme VDX 8770-4, and Extreme VDX 8770-8 devices.

To display MSTP information on the device in detail:

```
device# show spanning-tree mst detail
Spanning-tree Mode: Multiple Spanning Tree Protocol
CIST Root Id: 8000.0005.1e76.1aa0 (self)
CIST Bridge Id: 8000.0005.1e76.1aa0
CIST Reg Root Id: 8000.0005.1e76.1aa0 (self)
CIST Root Forward Delay: 15; Hello Time: 2; Max Age: 20; Max-hops: 20
Configured Forward Delay: 15; Hello Time: 2; Max Age: 20; Max-hops: 20;
Tx-HoldCount: 6
Number of topology change(s): 0
Bpdu-guard errdisable timeout: disabled
Bpdu-guard errdisable timeout interval: 300 sec
Migrate Time: 3 sec
```

CIST Port Details.

=====

Instance: 0; Vlans:1, 100

Port Te 0/0 enabled

```
IfIndex: 67108864; Id: 8000; Role: Disabled; State: Discarding
Designated External Path Cost: 0; Internal Path Cost 0
Configured Path Cost: 2000
Designated Port Id: 0; CIST Priority: 128
Designated Bridge: 0000.0000.0000.0000
CIST Port Hello Time: 2
Number of forward-transitions: 0
Version Multiple Spanning Tree Protocol - Received None - Send MSTP
Edgeport: off; AutoEdge: no; AdminEdge: no; EdgeDelay: 3 sec
Configured Root guard: off; Operational Root guard: off
Boundary: yes
Bpdu-guard: off
Bpdu-filter: off
Link-type: point-to-point
Received BPDUs: 0; Sent BPDUs: 0
```

Port Te 1/0/8 enabled

```
IfIndex: 67633408; Id: 8008; Role: Disabled; State: Discarding
Designated External Path Cost: 0; Internal Path Cost 0
Configured Path Cost: 2000
Designated Port Id: 0; CIST Priority: 128
Designated Bridge: 0000.0000.0000.0000
CIST Port Hello Time: 2
Number of forward-transitions: 0
Version Multiple Spanning Tree Protocol - Received None - Send MSTP
Edgeport: off; AutoEdge: no; AdminEdge: no; EdgeDelay: 3 sec
Configured Root guard: off; Operational Root guard: off
Boundary: yes
Bpdu-guard: off
Bpdu-filter: off
Link-type: point-to-point
Received BPDUs: 0; Sent BPDUs: 0
```

Port Te 1/0/19 enabled

```
IfIndex: 68354563; Id: 8013; Role: Disabled; State: Discarding
Designated External Path Cost: 0; Internal Path Cost 0
Configured Path Cost: 2000
Designated Port Id: 0; CIST Priority: 128
Designated Bridge: 0000.0000.0000.0000
CIST Port Hello Time: 2
Number of forward-transitions: 0
Version Multiple Spanning Tree Protocol - Received None - Send MSTP
Edgeport: off; AutoEdge: no; AdminEdge: no; EdgeDelay: 3 sec
Configured Root guard: off; Operational Root guard: off
Boundary: yes
Bpdu-guard: off
Bpdu-filter: off
Link-type: point-to-point
Received BPDUs: 0; Sent BPDUs: 0
```

Port Te 1/0/20 enabled

```
IfIndex: 68420100; Id: 8014; Role: Disabled; State: Discarding
```

show spanning-tree mst detail

```
Designated External Path Cost: 0; Internal Path Cost 0
Configured Path Cost: 2000
Designated Port Id: 0; CIST Priority: 128
Designated Bridge: 0000.0000.0000.0000
CIST Port Hello Time: 2
Number of forward-transitions: 0
Version Multiple Spanning Tree Protocol - Received None - Send MSTP
Edgeport: off; AutoEdge: no; AdminEdge: no; EdgeDelay: 3 sec
Configured Root guard: off; Operational Root guard: off
Boundary: yes
Bpdu-guard: off
Bpdu-filter: off
Link-type: point-to-point
Received BPDUs: 0; Sent BPDUs: 0
```

```
MSTI details.
=====
```

## Related Commands

[show spanning-tree mst instance](#), [show spanning-tree mst interface](#)

# show spanning-tree mst instance

Displays information on a specified Multiple Spanning Tree Protocol (MSTP) instance.

## Syntax

```
show spanning-tree mst instance instance_id [ interface port-channel number | interface <N>gigabitethernet rbridge-id/slot/
port ]
```

## Parameters

*instance\_id*

Specifies the MSTP instance for which to display information. Valid values range from 1 through 31.

**interface**

Specifies the interface for which to display the MSTP instance information.

**port-channel** *number*

Specifies the port-channel of the interface. The number of available channels range from 1 through 6144.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **gigabitethernet** *rbridge-id/slot/port* parameter is used only on Extreme VDX 6710, Extreme VDX 8770-4, and Extreme VDX 8770-8 switches.

## Examples

To display information on MSTP instance 1:

```
device# show spanning-tree mst instance 1 interface tengigabitethernet 1/0/0

Instance: 1; VLANs: 100
MSTI Root Id: 8001.0005.1e76.1aa0 (self)
MSTI Bridge Id: 8001.0005.1e76.1aa0
MSTI Bridge Priority: 32768
```

show spanning-tree mst instance

## Related Commands

[show spanning-tree mst brief](#), [show spanning-tree mst interface](#)



# show spanning-tree mst interface

Displays information for a specified interface for a Multiple Spanning Tree Protocol (MSTP) instance.

## Syntax

```
show spanning-tree mst interface [ port-channel number | <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

**port-channel** *number*

Specifies the port-channel of the interface. The number of available channels range from 1 through 6144.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display MSTP protocol specific information such as Common and Internal Spanning Tree (CIST) spanning-tree related information, information to each MSTP instance (MSTI), and the state of the port specific to each MSTI.

The **gigabitethernet** *rbridge-id/slot/port* parameter is used only on Extreme VDX 6710, VDX 8770-4, and VDX 8770-8 devices.

show spanning-tree mst interface

To display information for the MSTP interface:

```
device# show spanning-tree mst interface tengigabitethernet 5/0/1
Spanning-tree Mode: Multiple Spanning Tree Protocol
CIST Root Id: 8000.0005.1e76.1aa0 (self)
CIST Bridge Id: 8000.0005.1e76.1aa0
CIST Reg Root Id: 8000.0005.1e76.1aa0 (self)
IST Operational Port Hello Time: 0
Number of forward-transitions: 0
Version: Multiple Spanning Tree Protocol - Received None - Send MSTP
Edgeport: off; AutoEdge: no; AdminEdge: no; EdgeDelay: 3 sec
Configured Root guard: off; Operational Root guard: off
Boundary: yes
Bpdu-guard: off
Bpdu-filter: off
Link-type: point-to-point
Received BPDUs: 0; Sent BPDUs: 0
Instance Role   Sts   Cost   Prio VLANs
-----
0                DIS    DSC    2000    128                1
```

## Related Commands

[show spanning-tree brief](#), [show spanning-tree mst brief](#)

# show ssh server status

Displays the current Secure Shell (SSH) server key-exchange status.

## Syntax

```
show ssh server status [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### rbridge-id

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

When the SSH server is disabled on RBridge 3:

```
device# show ssh server status rbridge-id 3
rbridge-id 3 SSH server status: Enabled
```

When the SSH server key-exchange method is configured to DH Group 14:

```
device# show ssh server status rbridge-id 3
rbridge-id 3
SSH Kex Exchange Algorithm: DH Group 14
```

When the SSH server key-exchange method is restored to the default:

```
device# show ssh server status rbridge-id 3
rbridge-id 3
```

To display the number of maximum SSH sessions configured, as well as the status of VRFs, on RBridge 176:

```
device# show ssh server status rbridge-id 176
rbridge-id 176:
VRF-name: mgmt-vrf      Status: Enabled
VRF-name: default-vrf  Status: Enabled
rbridge-id 176: SSH Server Max sessions: 7
```

## History

Release version	Command history
7.1.0	This command was modified to include the status of VRFs and the number of SSH sessions configured.

show ssh server status

## Related Commands

[show telnet server status](#), [ssh](#), [ssh server shutdown](#)

# show ssh server rekey-interval status

Displays the status information related to the Secure Shell (SSH) server rekey-interval.

## Syntax

```
show ssh server rekey-interval status
```

## Modes

Privileged EXEC mode

# show startup-config

Displays the contents of the startup configuration.

## Syntax

**show startup-config**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local device.

## Examples

To display the startup configuration:

```
device# show startup-config

chassis virtual-ip 10.24.73.50/20
no diag post enable
linecard 2 LC48x10G
linecard 4 LC48x10G
class-map default
match any
!
logging rbridge-id 1
raslog console INFO
!
logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
logging syslog-facility local LOG_LOCAL7
switch-attributes 1
chassis-name VDX8770-4
host-name sw0
!
support rbridge-id 1
ffdc
!
snmp-server contact "Field Support."
snmp-server location "End User Premise."
snmp-server sys-descr "VDX device."
snmp-server community ConvergedNetwork
snmp-server community OrigEquipMfr rw
snmp-server community "Secret C0de" rw
snmp-server community common!
(Output truncated)
```

## Related Commands

[show running-config](#), [show startup-db](#)

# show startup-db

Displays the startup database information.

## Syntax

```
show startup-db
```

## Parameters

Refer to the Usage Guidelines.

## Modes

Privileged EXEC mode

## Usage Guidelines

Enter **show startup-db ?** to display the list of available database entries.

## Examples

To display the logging configuration in the startup database:

```
switch# show startup-db logging

logging rbridge-id 1
  raslog console INFO
!
logging auditlog class SECURITY
logging auditlog class CONFIGURATION
logging auditlog class FIRMWARE
logging syslog-facility local LOG_LOCAL7
```

## Related Commands

[show running-config](#), [show startup-config](#)

## show statistics access-list

For a given network protocol and inbound/outbound direction, displays ACL statistical information. You can show statistics for a specified ACL or only for that ACL on a specified interface. You can also display statistical information for all ACLs bound to a specified switch interface, VLAN, VE, or VXLAN overlay gateway. You can also display statistical information for receive-path ACLs on a specified RBridge or on all RBridges.

### Syntax

The following version displays statistical information for either the inbound or the outbound direction of a specified ACL:

```
show statistics access-list { ip | ipv6 | mac } name { in | out }
```

For either the inbound or the outbound direction on a specified N-gigabit physical Ethernet, port-channel, or VLAN interface, the following version displays statistical information for all ACLs bound to that interface:

```
show statistics access-list interface { <N>gigabitethernet rbridge_id/slot/port | port-channel index | vlan vlan_id } { in | out }
```

For either the inbound or the outbound direction on a specified virtual Ethernet (VE) interface, the following version displays statistical information for all ACLs bound to that interface. You can also include ACLs specific to an RBridge:

```
show statistics access-list interface ve vlan_id { in | out } [ rbridge-id { rbridge_id | all } ]
```

For the inbound direction on a specified VXLAN overlay-gateway, the following version displays statistical information for all ACLs bound to that gateway:

```
show statistics access-list overlay-gateway overlay_gateway_name in
```

For either the inbound or the outbound direction, on a specified N-gigabit physical Ethernet, port-channel, or VLAN interface, the following version displays statistics of the rules in a specified MAC ACL bound to that interface:

```
show statistics access-list mac name interface { <N>gigabitethernet rbridge_id/slot/port | port-channel index | vlan vlan_id } { in | out }
```

For either the inbound or the outbound direction, on a specified N-gigabit physical Ethernet or port-channel interface, the following version displays statistics of the rules in a specified Layer 3 ACL bound to that interface:

```
show statistics access-list { ip | ipv6 } name interface { <N>gigabitethernet rbridge_id/slot/port | port-channel index } { in | out }
```

For either the inbound or the outbound direction, on a specified virtual Ethernet (VE) interface, the following version displays statistics of the rules in a specified Layer 3 ACL bound to that interface. You can also include ACLs specific to an RBridge:

```
show statistics access-list { ip | ipv6 } name interface ve vlan_id in | out } [ rbridge-id { rbridge_id | all } ]
```

For a specified RBridge or for all RBridges, the following syntax displays statistics for a specified Layer 3 ACL applied to such RBridges.

```
show statistics access-list { ip | ipv6 } acl-name rbridge-id { rbridge_id | all } in
```

For a specified RBridge or for all RBridges, the following syntax displays statistics for one or both of the Layer 3 ACLs bound to such RBridges:

```
show statistics access-list rbridge-id { rbridge_id | all } in
```

### Parameters

**ip | ipv6 | mac**

Specifies the network protocol.

*name*

Specifies the ACL name.



**interface**

Filter by interface.

**<N>gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge\_id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel. Available channels range from 1 through 6144.

**vlan** *vlan\_id*

(Available only on Layer 2) Specifies a VLAN.

**ve** *vlan\_id*

(Available only on Layer 3) Specifies a virtual Ethernet (VE).

**rbridge-id**

Specifies one or all RBridges.

*rbridge\_id*

Specifies an RBridge.

**all**

(Available for VE only) Specifies all RBridges.

**overlay-gateway** *overlay\_gateway\_name*

Specifies a VXLAN overlay-gateway.

**in | out**

Specifies the ACL binding direction (incoming or outgoing).

## Modes

Privileged EXEC mode

## Usage Guidelines

Statistics are displayed only for ACL rules that contain the **count** keyword. This ACL keyword is not supported for this command on a management interface.

The ACL **log** keyword is not supported for this command on a management interface.

## Command Output

The **show statistics access-list** command displays the following information:

Output field	Description
Uncount	The counter resource is not allocated. This is typically seen if counting is not supported or if the hardware resources limit is reached.
Unwritten	The rule is inactive and is not programmed in the hardware. This is typically seen when the hardware resources limit is reached.

## Examples

The following example displays inbound ACL statistics for a named IPv4 ACL.

```
switch# show statistics access-list ip l3ext in
ip access-list l3ext TenGigabitEthernet 1/1/8 in
seq 76 deny ip 10.10.75.10 0.0.0.0 any count log (795239 frames)
seq 77 hard-drop ip 10.10.75.10 0.0.0.0 10.10.11.0 0.0.0.255 count log (0 frames)
seq 78 hard-drop ip any 10.10.11.0 0.0.0.255 count log (0 frames)
seq 79 hard-drop ip any 10.10.0.0 0.0.255.255 count log (0 frames)
seq 80 hard-drop ip 10.10.75.10 0.0.0.0 any count log (0 frames)
seq 81 hard-drop ip 10.10.75.0 0.0.0.0 10.10.0.0 0.0.255.255 count log (0 frames)
seq 91 hard-drop ip any any count (0 frames)
seq 100 deny udp 10.10.75.0 0.0.0.255 10.10.76.0 0.0.0.255 count log (0 frames)
seq 1000 permit ip any any count log (0 frames)
```

The following example displays inbound ACL statistics for a specified interface. The ACL named ipv6-std-acl is applied on interface 1/4/1 to filter incoming routed traffic only.

```
switch# show statistics access-list interface tengigabitethernet 1/4/1 in
ipv6 routed access-list ipv6-std-acl on TenGigabitEthernet 1/4/1 at Ingress (From User)
  seq 10 permit host 0::1
  seq 20 deny 0:2::/64
  seq 30 hard-drop any count (100 frames)
```

The following example displays inbound statistics for all ACLs bound to a specified VE interface.

```
switch# show statistics access-list interface ve 3010 in
ipv6 access-list ip_acl_3 on Ve 3010 at Ingress (From User)
  seq 10 deny ipv6 2001:3010:131:35::/64 2001:1001:1234:1::/64 count (0 frames)
  seq 20 permit ipv6 2001:3010:131:35::/64 2001:3001:1234:1::/64
```

The following example displays inbound statistics for all ACLs bound to an overlay gateway.

```
device# show statistics access-list overlay-gateway gw121 in
mac access-list stdmacaclin on overlay-gateway gw121 at Ingress (From User)
  seq 11 permit 1111.1112.1113 7777.7777.7777 count log (0 frames)
  seq 12 permit 1111.1112.1114 7777.7777.7777 count log (0 frames)

ip access-list stdipaclin on overlay-gateway gw121 at Ingress (From User)
  seq 11 deny 11.22.33.44 255.255.255.0 count log (0 frames)
  seq 12 deny 11.22.33.45 255.255.255.0 count log (0 frames)

ipv6 access-list stdipv6aclin on overlay-gateway gw121 at Ingress (From User)
  seq 20 deny any count log (0 frames)
```

The following example displays statistics for one or both of the Layer 3 ACLs applied to a specified RBridge.

```
device# show statistics access-list rbridge-id 1 in
ipv6 access-list ipv6-receive-acl-example on Rbridge 1 at Ingress (From Receive ACL)
  seq 10 hard-drop tcp host 10::1 any count (0 frames)
  seq 20 hard-drop udp any host 20::1 count (0 frames)
  seq 30 permit tcp host 10::2 any eq telnet count (0 frames)
  seq 40 permit tcp host 10::2 any eq bgp count (0 frames)
```

## History

Release version	Command history
7.1.0	The Usage Guidelines were modified to indicate that the ACL <b>count</b> and <b>log</b> keywords are not supported for this command on a management interface.

# show statistics rpf

Displays the number of packets dropped under unicast Reverse Path Forwarding (uRPF).

## Syntax

```
show statistics rpf
```

## Modes

Privileged EXEC mode

## Usage Guidelines

The uRPF drop counters are at ASIC level, not at interface level.

Incoming packets with DIP-lookup results in TRAP are lifted to the CPU—irrespective of uRPF check status—and are processed in the IP stack. Such packets that fail the uRPF check are dropped, but the uRPF drop-counter is not incremented.

## Command Output

The **show statistics rpf** command displays the following information:

Output field	Description
URPF drop count	Displays the number of packets dropped under uRPF.

## Examples

The following example displays the number of packets dropped under uRPF.

```
device# show statistics rpf
URPF drop count: 1970172
```

## History

Release version	Command history
7.2.0	This command was introduced.

# show storm-control

Displays all BUM (broadcast, unknown unicast and multicast)-related configurations in the system.

## Syntax

**show storm-control**

**show storm-control** [ **broadcast** | **multicast** | **unknown-unicast** ] [ **interface** { *<N>***gigabitethernet** } *rbridge-id/slot/port* ]

## Parameters

### **storm-control**

Displays all BUM-related configurations in the system.

### **broadcast**

Displays all BUM-related configurations in the system for the broadcast traffic type.

### **interface**

Displays all BUM-related configurations in the system for the specified interface.

### *<N>***gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace *<N>***gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

### **multicast**

Displays all BUM-related configurations in the system for the multicast traffic type.

### **unknown-unicast**

Displays all BUM-related configurations in the system for the unknown-unicast traffic type.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display BUM storm-control-related configuration for the entire system, for specified traffic types, for specified interfaces, or for specified traffic types on specified interface.

## Examples

To display storm control information for broadcast traffic on the 10-gigabit Ethernet interface 102/4/1:

```
device# show storm-control broadcast interface tengigabitethernet 102/4/1
```

Interface	Type	rate (Mbps)	conformed	violated	total
Tel02/4/1	broadcast	100,000	12500000000	12500000000	25000000000

To display storm control information for all traffic on the 10-gigabit Ethernet interface 102/4/1:

```
device# show storm-control interface tengigabitethernet 102/4/1
```

Interface	Type	rate (Mbps)	conformed	violated	total
Tel02/4/1	broadcast	100,000	12500000000	12500000000	25000000000
Tel02/4/1	unknown-unicast	100,000	12500000000	12500000000	25000000000
Tel02/4/1	multicast	100,000	12500000000	12500000000	25000000000

To display storm control information for all traffic in the system:

```
device# show storm-control
```

Interface	Type	rate (Mbps)	conformed	violated	total
Tel02/4/1	broadcast	100,000	12500000000	12500000000	25000000000
Tel02/4/1	unknown-unicast	100,000	12500000000	12500000000	25000000000
Tel02/4/1	multicast	100,000	12500000000	12500000000	25000000000
Tel02/4/2	broadcast	100,000	12500000000	12500000000	25000000000
Tel02/4/3	broadcast	100,000	12500000000	12500000000	25000000000
Tel02/4/4	unknown-unicast	100,000	12500000000	12500000000	25000000000

To display storm control information for all broadcast traffic the system:

```
device# show storm-control broadcast
```

Interface	Type	rate (Mbps)	conformed	violated	total
Tel02/4/1	broadcast	100,000	12500000000	12500000000	25000000000
Tel02/4/2	broadcast	100,000	12500000000	12500000000	25000000000
Tel02/4/3	broadcast	100,000	12500000000	12500000000	25000000000

# show support

Displays a list of core files on the switch.

## Syntax

```
show support [ rbridge-id { rbridge-id | all } ]
```

## Command Default

Displays information for the local switch.

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all switches in the fabric.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

Pagination is not supported with this command. Use the "more" parameter to display the output one page at a time.

## Examples

To display the core files:

```
switch# show support
```

```
No core or FFDC data files found!
```

## Related Commands

[clear support](#), [copy support](#)

# show system

Displays hardware and software system information.

## Syntax

```
show system [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

To display the system information:

```
device# show system

Stack MAC                : 00:05:33:4B:CC:37
  -- UNIT 0 --
Unit Name                 : sw0
Switch Status             : Online
Hardware Rev              : 97.4
Ten Gigabit Ethernet Port(s) : 60
Up Time                   : up 1 day, 2:29
Current Time              : 21:20:50 GMT
NOS Version               :
Jumbo Capable             : yes
Burned In MAC            : 00:05:33:4B:CC:37
Management IP            : 10.24.85.74
Management Port Status   : UP
  -- Power Supplies --
PS1 is faulty
PS2 is OK
  -- Fan Status --
Fan 1 is Ok
Fan 2 is Ok
Fan 3 is Ok
```

## Related Commands

[show interface stats](#), [show system monitor](#), [show system pstat interface](#), [show version](#)



# show system internal arp

Displays Address Resolution Protocol (ARP) information in the system.

## Syntax

```
show system internal arp interface ve vlan_id {in | out} [ rbridge-id {rbridge_id | all} ]
show statistics access-list interface { <N>gigabitethernet rbridge_id/slot/port | port-channel index | vlan vlan_id } { in | out }
  show system internal arp {I2 clientdb}
show system internal arp {clientlist structures}
show system internal arp {memstats}
show system internal arp {rib memstats}
show system internal arp {summary}
show system internal arp {vrf all}
```

## Parameters

**ip** | **ipv6** | **mac**

Specifies the network protocol.

*name*

Specifies the ACL name.

**interface**

Filter by interface.

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge\_id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel. Available channels range from 1 through 6144.

**vlan** *vlan\_id*

(Available only on Layer 2) Specifies a VLAN.

**ve** *vlan\_id*

(Available only on Layer 3) Specifies a virtual Ethernet (VE).

For either the inbound or the outbound direction on a specified virtual Ethernet (VE) interface, this option displays statistical information for all ACLs bound to that interface. You can also include ACLs specific to an RBridge.

**rbridge-id**

Specifies one or all RBridges.

*rbridge\_id*

Specifies an RBridge.

**all**

(Available for VE only) Specifies all RBridges.

**clientlist structures**

Displays the clientlist structures of the ARP information on the system.

**memstats**

Displays the memory statistics of the ARP information on the system.

**rib memstats**

Displays the RBridge memory statistics of the ARP information on the system.

**summary**

Displays a summary of the ARP information on the system.

*vrf all*

Displays all VRF statistics that relate to ARP on the system.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

Typical output for the **show system internal arp interface** command.

```

device# show system internal arp ipv4 vrf all
Idx : 7      IP : 2.1.1.1                MAC : 0027.f8c7.9778
Interface   : te0/12             Type      : MY-IP           Rib-Reg   : NO
MacResolved : YES                 Ivid     : 4099           L2-Port   : c018000
Flags       : 3                   ArpAgeTime: 265:25:54
-----
Idx : 19     IP : 2.2.2.2                MAC : 0027.f8c7.9769
Interface   : vlan0.100          Type      : MY-IP           Rib-Reg   : NO
MacResolved : NO                 Ivid     : 100          UNRES     : 22
Flags       : 1                   ArpAgeTime: 265:25:54
-----
Idx : 1      IP : 3.3.3.1                MAC : 0000.0000.0000
Interface   : lo1               Type      : MY-IP           Rib-Reg   : NO
MacResolved : NO                 Ivid     : 0            UNRES     : 0
Flags       : 1                   ArpAgeTime: 265:25:54
-----
Idx : 3      IP : 21.1.1.1              MAC : 0027.f8c7.9769
Interface   : vlan0.110         Type      : MY-IP           Rib-Reg   : NO
MacResolved : NO                 Ivid     : 110          UNRES     : 0
Flags       : 1                   ArpAgeTime: 265:25:54
-----
Idx : 8      IP : 30.1.1.31             MAC : 0005.33e6.caa7
Interface   : te0/23            Type      : DYNAMIC         Rib-Reg   : YES
MacResolved : YES                 Ivid     : 4098           L2-Port   : c02e000
Flags       : 7                   ArpAgeTime: 01:46:25
-----
Idx : 6      IP : 30.1.1.34             MAC : 0027.f8c7.9783
Interface   : te0/23            Type      : MY-IP           Rib-Reg   : NO
MacResolved : YES                 Ivid     : 4098           L2-Port   : c02e000
Flags       : 3                   ArpAgeTime: 265:25:54
-----
Idx : 2      IP : 100.1.1.1            MAC : 0027.f8c7.9769
Interface   : vlan0.100         Type      : MY-IP           Rib-Reg   : NO
MacResolved : NO                 Ivid     : 100          UNRES     : 0
Flags       : 1                   ArpAgeTime: 265:25:54
-----
Idx : 4      IP : 110.1.1.34           MAC : 0027.f8c7.9769
Interface   : vlan0.110         Type      : MY-IP           Rib-Reg   : NO
MacResolved : NO                 Ivid     : 110          UNRES     : 0
Flags       : 1                   ArpAgeTime: 265:25:54
-----
Idx : 9      IP : 110.1.1.120          MAC : 0010.9423.1201
Interface   : vlan0.110         Type      : DYNAMIC         Rib-Reg   : NO
MacResolved : NO                 Ivid     : 110          UNRES     : 22
Flags       : 1                   ArpAgeTime: 02:10:41
-----
Idx : 23     IP : 110.1.1.121          MAC : 0010.9423.1202
Interface   : vlan0.110         Type      : DYNAMIC         Rib-Reg   : NO
MacResolved : NO                 Ivid     : 110          UNRES     : 22
Flags       : 1                   ArpAgeTime: 02:10:41
-----

```

show system internal arp

Typical output for the **show system internal arp vrf all** command.

```
device# show system internal arp show vrf all
ARP Global Stats:
```

```
IPV4: Dynamic: 17      Static: 7      Evpn: 0      Sticky: 0      Pre: 0      Leak:
16      MyIP: 9
IPV6: Dynamic: 0      Static: 0      Evpn: 0      Sticky: 0      Pre: 0      Leak:
0      MyIP: 3      Linklocal: 0
```

```
-----
VrfId VrfName      AFI-ENABLE LeakCount  Dynamic  Static  BGP-Evpn  BGP-Sticky  Pre-
Arp    Leak-Arp    My-IP      V4/V6     V4/V6    V4/V6    V4/V6     V4/V6       V4/V6
V4/V6  V4/V6      V4/V6
-----
0      mgmt-vrf    1/1        0/0        0/0      0/0      0/0      0/0
0/0    0/0        0/0
1      default-vrf 1/1        1/0        17/0     7/0      0/0      0/0
0/0    0/0        8/3
2      red         1/1        0/0        0/0      0/0      0/0      0/0
0/0    16/0      0/0
3      blue        1/0        0/0        0/0      0/0      0/0      0/0
0/0    0/0        1/0
```

Typical output for the **show system internal arp summary** command.

```
device# show system internal arp show vrf all
Conversational ARP enabled: FALSE, AgeTime: 300
```

```
IPv4 ARP MAX: 16384    Current: 2      Exceeded: 0
IPv6 ND MAX: 4096     Current: 0      Exceeded: 0

IPV4: Dynamic: 2      Static: 0      Evpn: 0      Sticky: 0      Pre: 0      Leak:
0      MyIP: 2
IPV6: Dynamic: 0      Static: 0      Evpn: 0      Sticky: 0      Pre: 0      Leak:
0      MyIP: 0      Linklocal: 0
```

## History

Release version	Command history
7.0.1	This command was introduced.

# show system internal asic counter blk

Displays packet-count for block-level packet transfer between ASIC blocks.

## Syntax

```
show system internal asic counter blk { all | infed | intree | outfed | rxfifo | rxmcast | rxprs | swmfed | txqout }
```

## Parameters

**all**

Displays packet-count for packets transferred among all blocks.

**infed**

Displays the frame editor packet-in count.

**intree**

Displays packet-count for ingress on the routing block.

**outfed**

Displays the frame editor packet-out count.

**rxfifo**

Displays packet-count for packets transferred from the ingress port FIFO to the receive FIFO.

**rxmcast**

Displays packet-count for transfer to the multicast replication block.

**rxprs**

Displays packet-count for transfer from the receive FIFO to the parser.

**swmfed**

Displays packet-count for transfer to the frame editor.

**txqout**

Displays packet-count for transfer by the transmit queue block.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following example displays the packet count for all transfers among blocks.

```
device# show system internal asic counter blk all
Block Name                                     Packet Count
=====
Packets transferred from ingress port fifo to Receive Fifo : 7100669563
Packets transferred from receive fifo to parser           : 7100669563
Packets received at ingress on routing block              : 0
Packets transferred by the Tx queue block                 : 22839343553
Packets transferred to Frame editor                      : 23907883375
Frame editor packet in count                             : 0
Frame editor packet out count                            : 0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal asic counter drop-reason

Displays the count of dropped packets, sorted by with drop reasons.

## Syntax

```
show system internal asic counter drop-reason { all | rte | txq }
```

## Parameters

- all**  
Displays all drop-reason information.
- rte**  
Displays drop-counts from other blocks.
- txq**  
Displays drop counts from the transmit pipeline, with reasons.

## Modes

Privileged EXEC mode

## Usage Guidelines

Only non-zero values are displayed.

Dagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## History

Release version	Command history
5.0.0	This command was introduced.
7.0.0	This command was modified to support additional options.

# show system internal asic counter interface

Displays all nonzero MAC counters for a specified Ethernet interface.

## Syntax

```
show system internal asic counter interface <N>gigabitethernet rbridge-id / slot / port
```

## Parameters

<N>**gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example displays all nonzero MAC counters on a FortyGigabitEthernet interface.

```
device# show system internal asic counter interface FortyGigabitEthernet 9/0/21
ifm_if_update_ctrs - Unknown Counter type - type: 3   Interface FortyGigabitEthernet 9/0/21
  Db_slot : 0   RTE : 1   Asic_port : 60
===== Receive Ethernet Stats =====
Type                                     Hex      Dec
----                                     -
Total Packets                            4E       78
```



This example displays all nonzero MAC counters on a TenGigabitEthernet interface.

```

device# show system internal asic counter interface tengigabitethernet 53/0/25
Interface TenGigabitEthernet 68/0/25
  Db_slot : 0   RTE : 1   Asic_port : 70
===== Receive Ethernet Stats =====
Type                Hex      Dec
----              -
Total Packets       6990    27024
Packet size  64 - 127 bytes  5F75    24437
Packet size 128 - 255 bytes  73A     1850
Packet size 512 - 1023 bytes 2CC     716
Pause on Priority 4      15      21
Pause on Priority 5     5F75    24437
Pause on Priority 6     73A     1850

===== Transmit Ethernet Stats =====
Type                Hex      Dec
----              -
Total Packets       D8C     3468
Packet size 128 - 255 bytes  D8C     3468
Pause on Priority 0     6990    27024
Pause on Priority 1     D8C     3468
Pause on Priority 2     2D7DCF  2981327
Pause on Priority 3     C9828   825384
Pause on Priority 4     6990    27024

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal asic counter mem blk

Displays memory-error count for block-level packet transfer between ASIC blocks.

## Syntax

```
show system internal counter asic mem blk { all | ccb | epol | fed | ipol | l2 | l3 | mce | swm }
```

## Parameters

- all**  
Displays error-count from all blocks.
- ccb**  
Displays error-count from the congestion control block.
- epol**  
Displays error-count from the egress policer.
- fed**  
Displays error-count from the frame editor block.
- ipol**  
Displays error-count from the ingress policer block.
- l2**  
Displays error-count from the switching engine block.
- l3**  
Displays error-count from the routing engine block.
- mce**  
Displays error-count from the multicast replication block.
- swm**  
Displays error-count from the switch memory block.

## Modes

Privileged EXEC mode

## Usage Guidelines

Only non-zero values are displayed.

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following displays memory error-count from all blocks.

```
device# show system internal asic counter mem blk all
*****All Non-zero Memory/ECC Errors*****
Block Name          Error Type          Error Count
=====

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp evpn interface

Displays BGP EVPN debug information for interfaces for internal use.

## Syntax

```
show system internal bgp evpn interface { port-channel ifIndex-number | tunnel ifIndex-number }
```

## Parameters

**port-channel** *ifIndex-number*  
Specifies a port-channel interface.

**tunnel** *ifIndex-number*  
Specifies a tunnel interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows debug information for a specified port-channel interface in the system.

```
device# show system internal bgp evpn interface port-channel 671088641

IfName       : po1
IfType       : Port-Channel
Port Index   : 1
IfIndex      : 0x28000001
ESI          : 00.000000000000000000
ESI Derived  : None
ESR Originated : No
Link Status  : Down
LACP Priority : 0
LACP Partner Mac : 00:00:00:00:00:00
```

This example shows debug information for a specified tunnel interface in the system.

```
device# show system internal bgp evpn interface tunnel 2080436225

IfName       : te0/1
IfType       : Tunnel
Port Index   : 0
Dst IP       : 0x4e4e4e4e
Status      : 274007940
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp evpn l2route type

Displays debug information for BGP EVPN routes, filtered by route type, in the MAC-VRF table.

## Syntax

```
show system internal bgp evpn l2route type arp ip address mac mac address ethernet-tag tag-id
```

```
show system internal bgp evpn l2route type auto-discovery esi-value value ethernet-tag tag-id
```

```
show system internal bgp evpn l2route type ethernet-segment ethernet-tag tag-id { ipv4-address address | ipv6-address address }
```

```
show system internal bgp evpn l2route type inclusive-multicast ethernet-tag tag-id ipv4-address address [ l2-vni number
```

```
show system internal bgp evpn l2route type mac mac address ethernet-tag tag-id l2-vni number
```

```
show system internal bgp evpn l2route type nd IPv6 address mac mac address ethernet-tag tag-id
```

## Parameters

### type

Specifies a route type.

### arp

Specifies address-resolution protocol (ARP) routes.

### *ip address*

Specifies an IP address.

### mac *mac address*

Specifies Media Access Control (MAC) routes and specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

### ethernet-tag *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

### auto-discovery

Specifies automatically discovered routes.

### esi-value *value*

Specifies a 10 byte Ethernet Segment Identifier (ESI) value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH.HH).

### ethernet-segment

Specifies Ethernet Segment (ES) information.

### ipv4-address *address*

Specifies an IPv4 address.

### ipv6-address *address*

Specifies an IPv6 address.

### inclusive-multicast

Specifies inclusive multicast routes.

### l2-vni *number*

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

**nd**

Specifies neighbor-discovery (ND) routes.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output for the **show system internal bgp evpn l2route type** command when a specific MAC route is specified.

```
device# show system internal bgp evpn l2route type mac 0000.abba.abba ethernet-tag 11 l2-vni 11

EVPN NODE INFO:
-----
number_of_nlri_entries: 3, number_of_nlri_rejected: 0, already_added: 0, invalid: 0
recalc_pending: 0, delete_pending: 0, pending_processing: 0, redownload_pending: 0
vni_provisioned: 1, dampened: 0, num_moves: 0, last_source_local: 0
vpn: 0, last_update_timestamp: 0x0042c1ae, gen_id: 0, vlan_gen_id: 1

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: MAC:[11][0000.abba.abba], Status: BL, Age: 11h16m37s
  NEXT_HOP: 0.0.0.0, Learned from Peer: Local Router
  LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
  AS_PATH:
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
    Default Extd Gw Community: Received
    Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
    L2_vni: 11
    EST : 00.00000000000000000000
    RD: 9.0.0.9:1

NLRI:em:0000.abba.abba:11 Peer#: (2048) 255.255.255.255 state=6, address:0x12e7a894
tag:00000000
route_is B:1 M:0 L:1 installed:0 inactive:0 filtered:0, rm:0
route_is Is VPN:0 Rt_src:0 Rt_Type:0 exported:0
admin_distance:1 from_IBGP:0 from_Conf_EBGP:0 age:000009b2
route_is aggregate:0 suppressed:0 summary:0 Tree_node:0x12e80b66
As_path:0x12e24403 next_nlri:0x12e7a814, prev_nlri:0x12e7b114, magic 126
----- AS PATH Entry start-----
1 Next Hop : 0.0.0.0 MED :0 Origin:INCOMP
  Originator:0.0.0.0 Cluster List:None
  Aggregator:AS Number :0 Router-ID:0.0.0.0 Atomic:None
  Local Pref:100 Communities:Internet
  Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08
  AS Path : (length 0)
    AsPathLen: 0 AsNum: 0, SegmentNum: 0, Neighboring As: 0, Source As 0
  AsPath_Addr: 0x12e24403 Nh_Addr: 0x12e30f5c Nlri_Addr: 0x12e2d9e4 Hash:2957 (0x03000000)
  Links: 0x00000000, 0x00000000
  Reference Counts: 160:0:0, Magic: 126
----- AS PATH Entry ends -----
as_path_segments:0x12e3079e #AS:0(0), #Seq:0(0), Len:0
communities:0x12e24447 called:0, check_filter:0, no_export:0, no_advertise:0, local_as:0
neighboring_as: 0, source_as: 0, next-hop: 0.0.0.0, orig_as_path:0x00000000
----- Nexthop address start -----
Error: Next-hop 0.0.0.0 does not exist
----- Nexthop address ends -----
same bitLen next:0x262856c4, prev:0x00000000
aggregate:0x00000000, damping_nlri:00000000
RIB_out:0x00000000, #_RIB_out:0, weight:0
route_is restart_stale 0, route_is_special_network_local 0
waiting_for_label 0, ospf_ext_type2 0, ospf_rt_type 0, route_is_new 0, route_is_valid_nexthop 1
last_forwarding_route_modify_time 00000000, last_forwarding_route_modify_time_ack 00000000
Tx Path Id 0, Rx Path Id 0
2 Prefix: MAC:[11][0000.abba.abba], Status: E, Age: 10h5m46s
  NEXT_HOP: 78.0.0.78, Learned from Peer: 3.0.0.3 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 2 3
    Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT

3:11
  Default Extd Gw Community: Received
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  L2_vni: 11
```



ESI : 00.000000000000000000  
RD: 9.0.0.9:1

...

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp evpn l3vni

Displays debug information for BGP EVPN Layer 3 virtual network identifiers (VNIs) for internal use.

## Syntax

```
show system internal bgp evpn l3vni vrf name
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows BGP EVPN information for Layer 3 VNIs for VRF blue.

```
device# show system internal bgp evpn l3vni vrf blue
```

```
L3VNI RD tree Info
```

```
RD: 6.0.0.6:1
VNI: 20
mac: [0027.f8ca.76ba]
destination IP: 0x06000006
Vlan Installed: 1
Mac Installed: 1
Ref Count : 16
```

```
RD: 8.0.0.8:1
VNI: 20
mac: [50eb.1a13.cef5]
destination IP: 0x4e00004e
Vlan Installed: 1
Mac Installed: 1
Ref Count : 16
```

```
RD: 7.0.0.7:1
VNI: 20
mac: [50eb.1a14.0767]
destination IP: 0x4e00004e
Vlan Installed: 1
Mac Installed: 1
Ref Count : 16
```

```
RD: 6.0.0.6:2
VNI: 40
mac: [0027.f8ca.76ba]
destination IP: 0x06000006
Vlan Installed: 1
Mac Installed: 1
Ref Count : 16
```

```
RD: 8.0.0.8:2
VNI: 40
mac: [50eb.1a13.cef5]
destination IP: 0x4e00004e
Vlan Installed: 1
Mac Installed: 1
Ref Count : 16
```

```
RD: 7.0.0.7:2
VNI: 40
mac: [50eb.1a14.0767]
destination IP: 0x4e00004e
Vlan Installed: 1
Mac Installed: 1
Ref Count : 16
```

```
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp evpn neighbor

Displays BGP EVPN debug information about a specified neighbor for internal use.

## Syntax

```
show system internal bgp evpn neighbor { ip-addr | ipv6-addr }
```

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor.

*ipv6-addr*

Specifies the IPv6 address of a neighbor.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows information related to the processing of BGP EVPN for a specified neighbor

```

device# show system internal bgp evpn neighbor 2.0.0.2
index:2, base Address:12da5628
IP:2.0.0.2 (2.0.0.2) Remote_AS:2 (2) Enable:1 Shutdown:0 (generate-rib-out 0), 0
seconds_between_connection_attempts:10 current_seconds:10, up:0
LocalIP:9.0.0.9 Hold_timer:180 KeepAlive:60 Configured=0
safi: nego 0, adv 0, rcv 0, send_disable 0, enabled 1 (1)
GR: send_enable 0, sent 0, R_bit 0, this safi 0, Fbit 0, eor_s 0
    rcv restart 0, R_bit 0, restart_time 0, this safi 0, Fbit 0, eor_w 0
        stale_state 0, restarting_state 0
LAST: sent 0, R_bit 0, this safi 0, Fbit 0
    rcv restart 0, R_bit 0, restart_time 0, this safi 0, Fbit 0
GR Timers : purge_time_count 0, restart_time_count 0, staleroute_time_count 0 reStartStaleTimer 0
Fill Index: 0, Detach Index: 0
Weight:0 Local_Pref:100
Max_Pref=0 Threshold_Percent:0 Teardown:0
Warning_Limit=0 Temporary_Shutdown:0 Exceed_Warning_Limit:0
UpdateSouceInterface:14564, UpdateSoucePort:14564, ebgp_multihop:1 ttl:5, btsh: 0
maxas-limit:300
enforce-first-as:0
Password Encrypt Code:0, String:<>
Password length:0, String:<xxxxxx> Next_hop_self:0, always 0
SendCommunity:0, Send-label: 0, Remove_Private_as:0 Reflector_Client:0 Static_network_edge: 0
As_Override:0, Allowas_in count=5
Orig_Def:0, Orig_Def_Route-Map:<> 00000000
Default tx count:0 - 0, (0) rx count: 0 - 0
After changing additional-paths is clear pending: No
Add Path Capability enabled for safi: No
Add Path Capability Negotiation for sending: FALSE for receiving: FALSE
Route advertisement interval 0 seconds, last send at 43359 seconds ago, Peer ready to send updates
AS4 capability negotiated = 0
AS4 capability configured = 0
AsPathFilter In:0, Number:0 Out:0 Number:0
AddrFilter In:0, Number:0 Out:0 Number:0
PrefixList In:<>, 00000000 Out:<>, 00000000
AsPathACL In:, Out: CommunityACL In:, Out:
AddressPrefixOptionReceived: 0, AddressPrefixConfigOption: 0, AddressPrefixOptionNego: 0, orf
00000000
AllowedToSendUpdate: 0, SendPrefixOrfUpdate: 0, PrefixOrfSentCount: 0
ExtCommOrfOptionReceived: 0, ExtCommOrfConfigOption: 0, ExtCommOrfOptionNego: 0
SendExtcommOrfUPdate: 0, ExtcommOrfSentCount: 0
ExtComm Imported ORFs: []
IpACL In:0 Out:0
IpACLName In:<>, 00000000 Out:<>, 00000000
Current Remote_as:2 State:2 clearing safi 0, 0, 0, 0, 0, '*' 0/0/0, '+' 0
Tcp_close_pending :0
    Outbound Policy Group (safi 0): 0x12e6d3c0 (Hash 0), ID: 2, P#: 0, Drop 0, Use Count: 2,
Staring: 0, Update: 0, CCnt: 0
    Ribout Group: 0x208250d8, ID: 2, Type: 1, Peer Count: 2, Mask: 0x00000003 (1), ribout: 6,
withdrawn: 0
    Last update time was 43285 sec ago
    Outbound Policy Group (safi 7): 0x12db1398 (Hash 0), ID: 2, P#: 0, Drop 0, Use Count: 2,
Staring: 0, Update: 0, CCnt: 0
    Ribout Group: 0x2081a870, ID: 2, Type: 1, Peer Count: 2, Mask: 0x00000003 (1), ribout: 947,
withdrawn: 0
    Stale Route cal info : 0.0.0.0/0, next 0, schd 0
MsgHeaderLen:0 MsgLen:0 ErrorMessageRcvd:0 ErrorMessageLen:0
Last error
TCB:00000000, seq:00000000 (0) Port:14564 Port_state:0
Connection_up:0 Init_by_us:0 IBGP:0 Conf_EBGP:0
default_sent:0 timer_counter:0
Hold_Timer_En:0 timer_expire:0, current:867188 Diff: 214705005 sec NegotiatedValue:180
KeepAlive_Timer_En:0 Value:60 timer_expire:0, current:867188 Diff: 214705005 sec

```

show system internal bgp evpn neighbor

```
Connect_Retry_Timer_En:1 Value:20 Counter:0
error_enter_idle?:0 Reason:0
NLRI#:0 No of filtered NLRI for soft reconfig:0
Withdrawn_route#: (group 0) To_send 0
Rib_out#: (group 947) 947, To_send#:947
Counting: NLRI#:0 As_Path#:0
    Rib_out: (group 947) 947 To_Send:947 (947);    Withdrawn Route: (group 0) To_Send: 0 (0)
Rib_group change: in progress 0, pending in 0 sec
    As_path hash:max_as_path_cnt:0, max_nlri_cnt:0
PeerGrp:<> Neighbor Config:
remote_as:1
safi(7): activate: 1, Rib_out#:947 To_send#:947
Inbound policy cache not available
Outbound policy cache as_path 0x13fac3ae to 0x13fac3ae
----- BFD -----
Peer State : 0 Remote address 2.0.0.2 Outgoing interface 0
Bfd Session State 0 BFD-Down Count 0 BGP-BFd down Count 0
Holdover timer 0 Multiplier 0 Negotiated Rx 0 Negotiated Tx 0
Session-id 0
    Last-session-created 0 last-session-updated 0 last-session-deleted 0 last-session-down 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
    Event(0): 0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp evpn routes type

Displays debug information for BGP EVPN routes, filtered by route type, in the BGP EVPN table.

## Syntax

```
show system internal bgp evpn routes type arp ip address mac mac address ethernet-tag tag-id l2-vni number
```

```
show system internal bgp evpn routes type auto-discovery esi-value value ethernet-tag tag-id
```

```
show system internal bgp evpn routes type ethernet-segment ethernet-tag tag-id { ipv4-address address | ipv6-address address }
```

```
show system internal bgp evpn routes type inclusive-multicast ethernet-tag tag-id ipv4-address address [ l2-vni number
```

```
show system internal bgp evpn routes type ipv4-prefix ip address/mask tag tag-id l3vni value
```

```
show system internal bgp evpn routes type ipv6-prefix ipv6 address/mask tag tag-id l3vni value
```

```
show system internal bgp evpn routes type mac mac address ethernet-tag tag-id l2-vni number
```

```
show system internal bgp evpn routes type nd IPv6 address mac mac address ethernet-tag tag-id l2-vni number
```

## Parameters

### type

Specifies a route type.

### arp

Specifies address-resolution protocol (ARP).

### *ip address*

Specifies a route type.

### mac *mac address*

Specifies Media Access Control (MAC) information and specifies a MAC address. The valid format is HHHH.HHHH.HHHH.

### ethernet-tag *tag-id*

Specifies an Ethernet tag. Valid values range from 1 through 4294967295.

### l2-vni *number*

Specifies a layer 2 virtual network identifier (VNI). Valid values range from 1 through 16777215.

### auto-discovery

Specifies automatically discovered routes.

### esi-value *value*

Specifies a 10 byte Ethernet Segment Identifier (ESI) value in the form of hexadecimal characters (HH.HH.HH.HH.HH.HH.HH.HH.HH.HH).

### ethernet-segment

Specifies Ethernet Segment (ES) information.

### ipv4-address *address*

Specifies an IPv4 address.

show system internal bgp evpn routes type

**ipv6-address** *address*

Specifies an IPv6 address.

**inclusive-multicast**

Specifies inclusive multicast information.

**ipv4-prefix**

Specifies IPv4 prefix information.

*IPv4 address/mask*

Specifies an IPv4 address and mask.

**l3vni** *value*

Specifies a Layer 3 virtual network identifier (VNIs). Valid values range from 1 through 6777215.

**ipv6-prefix**

Specifies IPv6 prefix information.

*IPv6address/mask*

Specifies an IPv6 address and mask.

**nd**

Specifies neighbor-discovery (ND) information.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.



## Examples

This example shows sample output for the **show system internal bgp evpn route type** command when the **ipv4-prefix** keyword is used. Debug information for BGP EVPN IPv4 prefix routes is shown.

```
device# show system internal bgp evpn routes type ipv4-prefix 11.1.0.0/16 tag 0 l3vni 20

EVPN NODE INFO:
-----
number_of_nlri_entries: 1, number_of_nlri_rejected: 0, already_added: 0, invalid: 0
recalc_pending: 0, delete_pending: 0, pending_processing: 0, redownload_pending: 0
vni_provisioned: 0, dampened: 0, num_moves: 0, last_source_local: 0
vpn: 1, last_update_timestamp: 0x00000000, gen_id: 0, vlan_gen_id: 0

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: IP4Prefix:[0][11.1.0.0/16], Status: BE, Age: 10h22m0s
  NEXT_HOP: 6.0.0.6, Learned from Peer: 3.0.0.3 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
  AS_PATH: 2 3
    Extended Community: RT 1:1 ExtCom:06:03:00:27:f8:ca:76:ba ExtCom:03:0c:00:00:00:00:08

RT 3:20
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 1, Admin distance 20
  L3_vni: 20 Router Mac : 0027.f8ca.76ba
  RD: 6.0.0.6:1

NLRI:ep4:11.1.0.0/16:0 Peer#:(3) 3.0.0.3 state=6, address:0x2628f044
tag:00000000
route_is B:1 M:0 L:0 installed:1 inactive:0 filtered:0, rm:0
route_is Is_VPN:1 Rt_src:0 Rt_Type:0 exported:0
admin_distance:20 from_IBGP:0 from_Conf_EBGP:0 age:000155da
route_is aggregate:0 suppressed:0 summary:0 Tree_node:0x25b5fa41
As_path:0x12e27a4e next_nlri:0x2628efc4, prev_nlri:0x2628f0c4, magic 1270
----- AS PATH Entry start-----
1 Next Hop : 6.0.0.6 MED :0 Origin:INCOMP
  Originator:0.0.0.0 Cluster List:None
  Aggregator:AS Number :0 Router-ID:0.0.0.0 Atomic:None
  Local Pref:100 Communities:Internet
  Extended Community: RT 1:1 ExtCom:06:03:00:27:f8:ca:76:ba ExtCom:03:0c:00:00:00:00:08 RT

3:20
  AS Path :2 3 (length 3)
  AsPathLen: 2 AsNum: 2, SegmentNum: 1, Neighboring As: 2, Source As 3
  AsPath_Addr: 0x12e27a4e Nh Addr: 0x12e32ee8 Nlri_Addr: 0x2628ebc4 Hash:2179 (0x0300023a)
  Links: 0x00000000, 0x00000000
  Reference Counts: 16:0:16, Magic: 1270
----- AS PATH Entry ends -----
as_path_segments:0x12e88556 #AS:2(0), #Seq:1(0), Len:2
communities:0x12e27a92 called:0, check_filter:0, no_export:0, local_as:0
neighboring_as: 2, source_as: 3, next-hop: 6.0.0.6, orig_as_path:0x00000000
----- Nexthop address start -----
Next-hop 6.0.0.6, safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost
0, plen 0, number_of_paths 0
  last_update_time 0, resolve_source 0, reCalculate 0
  ribSubscribeState 0, ribSubscribe_time 00000000, ribResolved_time 00000000
----- Nexthop address ends -----
same bitLen next:0x00000000, prev:0x00000000
aggregate:0x00000000, damping_nlri:00000000
RIB_out:0x262e4b7c, #_RIB_out:1, weight:0
route_is restart stale 0,route_is_special_network_local 0
waiting_for_label 0, ospf_ext_type2 0, ospf_rt_type 0, route_is_new 0, route_is_valid_nexthop 1
last_forwarding_route_modify_time 00000000, last_forwarding_route_modify_time_ack 00000000
Tx Path Id 0, Rx Path Id 0
2 Prefix: IP4Prefix:[0][11.1.0.0/16], Status: BE, Age: 10h22m2s
  NEXT_HOP: 78.0.0.78, Learned from Peer: 3.0.0.3 (2)
  LOCAL_PREF: 100, MED: none, ORIGIN: incomplete, Weight: 0
```

show system internal bgp evpn routes type

```
AS_PATH: 2 3
  Extended Community: RT 1:1 ExtCom:06:03:50:eb:1a:13:ce:f5 ExtCom:03:0c:00:00:00:00:08
RT 3:20
  Extended Community: ExtCom: Tunnel Encapsulation (Type Vxlan)
  Adj_RIB_out count: 1, Admin distance 20
  L3_vni: 20 Router Mac : 50eb.1a13.cef5
  RD: 8.0.0.8:1
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp evpn variables

Displays debug information about the BGP EVPN internal system variables for internal use.

## Syntax

```
show system internal bgp evpn variables
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output for the **show system internal bgp evpn variables** command.

```
device# show system internal bgp evpn variables

*** show system internal bgp evpn variables safi(7) vrfId(1) ****
safi:7, &bgp:3fa75058, enabled:1, operational:1, trace_dbg_mem =0, curr_afi:1 vr_init:1
io_process_running:0, io_process_next_peer_number=0
in_long_loops 0, clear_all 0, timer 00000000, count 0
timer_enabled:1, timer_next_peer_number:3, 1s timer 1, short timer 1
scheduler id:0:0, ip:0.0.0.0/0, time=11310272
bgp_tcb:10761710 (0x00000000, 67), tick_cnt=0, seconds=14
bgp_tcb6:3fa74560 (0x00000000, 0x00000043)
Auto shutdown new neighbor enabled= 0
*peer:1076330c, *peer_group:107874c0, RIB_in_root_node:00000000
Maximum Peer Index Number:4, check_nexthops:0 0
router_id:9.0.0.9, configured:1, cluster_id:0.0.0.0, configured:0
route_is_router_reflector:0, client_to_client_reflection:1
networks:x00000000, aggregate:x00000000
default_metric:0, local_preference:100, keep_alive:60, hold_time:180
originate_default:0, originated:0
vr_originate_default_ribSubscribeState 0, vr_originate_default_ribSubscribe_time 00000000,

vr_originate_default_ribUnSubscribe_time 00000000, vr_originate_default_ribResolved_time 00000000
Rib Route Count (v4:4 v6:6)
Ack_pending (v4:0 v6:0)
distance:20 200 200, fast_external_fallover=0
nexthop recur1, nexthop recur ipv6 1, en_def:0, readvertise:1, auto_sum:0, synch:0
always_compare_med:0, compare_med_with_empty_aspath: 0, redistribute_ibgp:0,

local_network_check_time_count:563715
vr_maximum_runtime_ip_paths[i/e]:1, 1, vr_maximum_runtime_bgp_paths[i/e]:1, 1
nexthop_cache_hit_Count:0, nexthop_cache_miss_count:28
system memory:268848644, available:0; total_allocated:4802274, bgp_defined_quota:500000000
Memory 85Reached: 0, Memory 90Reached 0
v4 import map:"", 0x00000000 v4 export map:"", 0x00000000
v6 import map:"", 0x00000000 v6 export map:"", 0x00000000
v4_nexthop_lb_interface:<no-such-port>, v6_nexthop_lb_interface:<no-such-port>,

v4_nexthop_lb_addr:0.0.0.0
v6_nexthop_lb_addr:0.0.0.0
vrf instances:4
 [safi:7][type:0].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:1].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:2].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:3].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:4].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:5].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:6].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:7].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:7][type:8].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
*** show system internal bgp evpn GR safi(7) vrfId(1) ****
```

```

BGP Debug Information
Pid Size      Address Total      Used      Free      NoMem      Errors      #_pools      p_unit
0   8      12e140d8 252      107      145      0          0          2          100
1   16     1f6f2ef8 678      272      406      0          0          3          100
2   24     2340b790 765      588      177      0          0          3          100
3   32     12d16770 50       5        45       0          0          1          40
4   48     1f6f55d0 724      574      150      0          0          7          20
5   64     12dealf8 278      270      8        0          0          7          10
6   96     2531deb0 311      227      84       0          0          6          10
7   128    23416eb8 117      77       40       0          0          4          10
8   256    00000018 0        0        0        0          0          0          10
9   68     1f6eea30 215      145      70       0          0          1          200
10  60     24025080 12525    380      12145    0          0          4          500
11  120    136e2588 6648     1912     4736    0          0          8          200
12  115    234431b8 1064     588      476     0          0          3          200
13  40     00000018 0        0        0        0          0          0          10
14  32     00000018 0        0        0        0          0          0          10
15  2824   1f636620 11       4        7        0          0          1          10
16  105968 232a7020 18       12       6        0          0          2          10
17  49704  23031020 10       6        4        0          0          1          10
18  90     12dc1808 10       3        7        0          0          1          10
19  74     12de9220 440      432      8        0          0          11         10
20  79     23c64da0 367      200      167     0          0          10         10
21  91     12de5248 368      304      64      0          0          11         10
22  75     00000018 0        0        0        0          0          0          10
23  87     00000018 0        0        0        0          0          0          10
24  73     1f6d9ab0 198      186      12      0          0          6          10
25  85     00000018 0        0        0        0          0          0          10
26  73     12de6248 174      136      38      0          0          5          10
27  85     26d4aba8 170      136      34      0          0          6          10
28  665    1f6b6168 194      16       178     0          0          1          100
29  75     26dd9b58 489      448      41      0          0          12         10
30  85     234421f8 172      136      36      0          0          4          10
31  4096   00000018 0        0        0        0          0          0          10
32  5120   00000018 0        0        0        0          0          0          10

Total Memory Use for bgp memory pools : 4779804
Memory Block Not Available Count : 0
Bad Memory Pool ID Count : 0
Route Calculation info : 0.0.0.0/0, next 0 sch 0
BGP route update count : 0 (0) last:
event : (0:0) 0.0.0.0/0

MP active: 1, standby up 1
Graceful_restart: enable 1, restart time 120, stale-route 360, purge 600
Restarted 0, fwd 1, restart_up_time_count 0, safi 7, igpSwitchoverPending 0 vrNhSyncDoneSent 0

ribGrDoneRcvd:1

BGP inbound/outbound policy caching Enabled.
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp evpn vlan-db

Displays BGP EVPN debug information for VLAN databases for internal use.

## Syntax

```
show system internal bgp evpn vlan-db [ detail | gvid number | ifindex number | vni number
```

## Parameters

### detail

Specifies detailed information for VLAN databases in the system.

### gvid *number*

Specifies .

### ifindex *number*

Specifies .

### vni *number*

Specifies a virtual network identifier (VNI). Valid values range from 1 through 16777215.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output for the **show system internal bgp evpn vlan-db** command when the **detail** keyword is used.

```
device# show system internal bgp evpn vlan-db detail

IfName      : vlan0.1
IVID        : 1
GVID        : 1
Port Index  : 6272
IfIndex     : 0x48000001
Extented    : 1
VNI         : 1
ARPD Regis  : TRUE
ARP AFv4/6  : FALSE/FALSE
L2S Regis   : TRUE
L3 VNI      : 1
If Mac      : 0x00000000:0x00000000:0x00000000:0x00000000:0x00000000:0x00000000

IfName      : vlan0.11
IVID        : 11
GVID        : 11
Port Index  : 6282
IfIndex     : 0x4800000b
Extented    : 1
VNI         : 11
ARPD Regis  : TRUE
ARP AFv4/6  : TRUE/TRUE
L2S Regis   : TRUE
L3 VNI      : 4
If Mac      : 0x00000000:0x00000027:0x000000f8:0x000000fd:0x00000027:0x0000004b

IfName      : vlan0.12
IVID        : 12
GVID        : 12
Port Index  : 6283
IfIndex     : 0x4800000c
Extented    : 1
VNI         : 12
ARPD Regis  : TRUE
ARP AFv4/6  : TRUE/TRUE
L2S Regis   : TRUE
L3 VNI      : 4
If Mac      : 0x00000000:0x00000027:0x000000f8:0x000000fd:0x00000027:0x0000004b

IfName      : vlan0.13
IVID        : 13
GVID        : 13
Port Index  : 6284
IfIndex     : 0x4800000d
Extented    : 1
VNI         : 13
ARPD Regis  : TRUE
ARP AFv4/6  : TRUE/TRUE
L2S Regis   : TRUE
L3 VNI      : 4
If Mac      : 0x00000000:0x00000027:0x000000f8:0x000000fd:0x00000027:0x0000004b

IfName      : vlan0.14
IVID        : 14
GVID        : 14
Port Index  : 6285
IfIndex     : 0x4800000e
Extented    : 1
VNI         : 14
ARPD Regis  : TRUE
ARP AFv4/6  : TRUE/TRUE
L2S Regis   : TRUE
L3 VNI      : 4
If Mac      : 0x00000000:0x00000027:0x000000f8:0x000000fd:0x00000027:0x0000004b
```

```
show system internal bgp evpn vlan-db
```

```
IfName      : vlan0.15
IVID       : 15
GVID       : 15
Port Index  : 6286
IfIndex     : 0x4800000f
Extented    : 1
VNI        : 15
ARPD Regis : TRUE
ARP AFv4/6 : TRUE/TRUE
L2S Regis  : TRUE
L3 VNI     : 4
If Mac     : 0x00000000:0x00000027:0x000000f8:0x000000fd:0x00000027:0x0000004b

IfName      : vlan0.16
IVID       : 16
GVID       : 16
Port Index  : 6287
IfIndex     : 0x48000010
Extented    : 1
VNI        : 16
ARPD Regis : TRUE
ARP AFv4/6 : TRUE/TRUE
L2S Regis  : TRUE
L3 VNI     : 4
If Mac     : 0x00000000:0x00000027:0x000000f8:0x000000fd:0x00000027:0x0000004b
...
```

This example shows sample output for the **show system internal bgp evpn vlan-db** command when the **gvid** keyword is used.

```
device# show system internal bgp evpn vlan-db gvid 16
IfName      : vlan0.16
IVID       : 16
GVID       : 16
Port Index  : 6287
IfIndex     : 0x48000010
Extented    : 1
VNI        : 16
ARPD Regis : TRUE
ARP AFv4/6 : TRUE/TRUE
L2S Regis  : TRUE
L3 VNI     : 4
If Mac     : 0x00000000:0x00000027:0x000000f8:0x000000fd:0x00000027:0x0000004b
```

## History

Release version	Command history
7.0.0	This command was introduced.



# show system internal bgp ipv4 config

Displays debug information related to the processing of BGP4 configurations for internal use.

## Syntax

```
show system internal bgp ipv4 config [ vrf name ]
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows BGP4 configuration information.

```

device# show system internal bgp ipv4 config

*** show running-config vrf config ***
vrf blue
  vni 40
  vrf rd 9.0.0.9:2
  route-target export RT 2:2 ,
  route-target import RT 2:2 ,
  route-target export RT 2:2 ,
  route-target import RT 2:2 ,
vrf green
  vni 60
  vrf rd 9.0.0.9:3
  route-target export RT 3:3 ,
  route-target import RT 3:3 ,
  route-target export RT 3:3 ,
  route-target import RT 3:3 ,
vrf red
  vni 20
  vrf rd 9.0.0.9:1
  route-target export RT 1:1 ,
  route-target import RT 1:1 ,
  route-target export RT 1:1 ,
  route-target import RT 1:1 ,
vrf yellow
  vni 80
  vrf rd 9.0.0.9:4
  route-target export RT 4:4 ,
  route-target import RT 4:4 ,
  route-target export RT 4:4 ,
  route-target import RT 4:4 ,

*** show running-config bgp config ***
evpn-instance
  rd auto
  route-target export auto
  route-target import auto
  vni add 1-18, 21-28, 41-48, 61-68, 81-88
!
local-as 3
neighbor 2.0.0.2 remote-as 2
neighbor 2.0.0.2 ebgp-multihop 5
neighbor 2.0.0.2 update-source Loopback 1
neighbor 3.0.0.3 remote-as 2
neighbor 3.0.0.3 ebgp-multihop 5
neighbor 3.0.0.3 update-source Loopback 1
neighbor 1000:2:9:: remote-as 2
neighbor 1000:3:9:: remote-as 2

address-family ipv4 unicast
  graceful-restart
  maximum-paths 8
  redistribute connected
  neighbor 2.0.0.2 allowas-in 5
  neighbor 3.0.0.3 allowas-in 5
  no neighbor 1000:2:9:: activate
  no neighbor 1000:3:9:: activate
  next-hop-recursion
  exit-address-family

address-family ipv4 multicast
  next-hop-recursion
  exit-address-family

address-family ipv6 unicast
  graceful-restart
  maximum-paths 8

```

```

redistribute connected
neighbor 1000:2:9:: activate
neighbor 1000:2:9:: allowas-in 5
neighbor 1000:3:9:: activate
neighbor 1000:3:9:: allowas-in 5
next-hop-recursion
exit-address-family

address-family l2vpn evpn
graceful-restart
neighbor 2.0.0.2 activate
neighbor 2.0.0.2 allowas-in 5
neighbor 2.0.0.2 next-hop-unchanged
neighbor 2.0.0.2 send-community extended
neighbor 3.0.0.3 activate
neighbor 3.0.0.3 allowas-in 5
neighbor 3.0.0.3 next-hop-unchanged
neighbor 3.0.0.3 send-community extended
next-hop-recursion
exit-address-family

address-family ipv4 unicast vrf blue
maximum-paths 8
redistribute connected
exit-address-family

address-family ipv6 unicast vrf blue
redistribute connected
exit-address-family

address-family ipv4 unicast vrf green
maximum-paths 8
redistribute connected
exit-address-family

address-family ipv6 unicast vrf green
redistribute connected
exit-address-family

address-family ipv4 unicast vrf red
maximum-paths 8
redistribute connected
exit-address-family

address-family ipv6 unicast vrf red
redistribute connected
exit-address-family

address-family ipv4 unicast vrf yellow
maximum-paths 8
redistribute connected
exit-address-family

address-family ipv6 unicast vrf yellow
redistribute connected
exit-address-family

*** show running-config bgp config End ****

```

This example shows BGP4 configuration information for VRF "yellow".

```

device# show system internal bgp ipv4 config vrf yellow

address-family ipv4 unicast vrf yellow
maximum-paths 8
redistribute connected
exit-address-family

```

show system internal bgp ipv4 config

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv4 neighbor

Displays BGP4-related debug info for the specified neighbor.

## Syntax

```
show system internal bgp ipv4 network ip-address [ vrf name ]
```

## Parameters

*ip-address*

Specifies an IP address.

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv4 neighbor** command.

```

device# show system internal bgp ipv4 neighbor 3.0.0.3
index:3, base Address:12dab6c0
IP:3.0.0.3 (3.0.0.3) Remote_AS:2 (2) Enable:1 Shutdown:0 (generate-rib-out 0), 0
seconds_between_connection_attempts:20 current_seconds:141645, up:1
LocalIP:9.0.0.9 Hold_timer:180 KeepAlive:60 Configured=0
safi: nego 1, adv 1, rcv 1, send_disable 0, enabled 1 (1)
GR: send_enable 1, sent 1, R_bit 0, this safi 1, Fbit 0, eor_s 0
    rcv restart 1, R_bit 0, restart_time 120, this safi 1, Fbit 0, eor_w 0
        stale_state 0, restarting_state 0
LAST: sent 0, R_bit 0, this safi 0, Fbit 0
    rcv restart 0, R_bit 0, restart_time 0, this safi 0, Fbit 0
GR Timers : purge_time_count 0, restart_time_count 0, staleroute_time_count 0 reStartStaleTimer 0
Fill Index: 8, Detach Index: 8
Weight:0 Local_Pref:100
Max_Pref=0 Threshold_Percent:0 Teardown:0
Warning_Limit=0 Temporary_Shutdown:0 Exceed_Warning_Limit:0
UpdateSouceInterface:14564, UpdateSoucePort:14564, ebgp_multihop:1 ttl:5, btsh: 0
maxas-limit:300
enforce-first-as:0
Password Encrypt Code:0, String:<>
Password length:0, String:<xxxxxx> Next_hop_self:0, always 0
SendCommunity:0, Send-label: 0, Remove_Private_as:0 Reflector_Client:0 Static_network_edge: 0
As_Override:0, Allowas_in count=5
Orig_Def:0, Orig_Def_Route-Map:<> 00000000
Default tx count:0 - 0, (0) rx count: 0 - 0
After changing additional-paths is clear pending: No
Add Path Capability enabled for safi: No
Add Path Capability Negotiation for sending: FALSE for receiving: FALSE
Route advertisement interval 0 seconds, last send at 1259 seconds ago, Peer ready to send updates
AS4 capability negotiated = 0
AS4 capability configured = 0
AsPathFilter In:0, Number:0 Out:0 Number:0
AddrFilter In:0, Number:0 Out:0 Number:0
PrefixList In:<>, 00000000 Out:<>, 00000000
AsPathACL In:, Out: CommunityACL In:, Out:
AddressPrefixOptionReceived: 0, AddressPrefixConfigOption: 0, AddressPrefixOptionNego: 0, orf
00000000
AllowedToSendUpdate: 1, SendPrefixOrfUpdate: 0, PrefixOrfSentCount: 0
ExtCommOrfOptionReceived: 0, ExtCommOrfConfigOption: 0, ExtCommOrfOptionNego: 0
SendExtcommOrfUPdate: 0, ExtcommOrfSentCount: 0
ExtComm Imported ORFs: []
IpACL In:0 Out:0
IpACLName In:<>, 00000000 Out:<>, 00000000
Current Remote_as:2 State:6 clearing safi 0, 0, 0, 0, 0, '*' 0/0/0, '+' 0
Tcp_close_pending :0
    Outbound Policy Group (safi 0): 0x12e6d3c0 (Hash 0), ID: 2, P#: 0, Drop 0, Use Count: 2,
Staring: 0, Update: 0, CCnt: 0
    Ribout Group: 0x208250d8, ID: 2, Type: 1, Peer Count: 2, Mask: 0x00000003 (1), ribout: 6,
withdrawn: 0
    Last update time was 146343 sec ago
    Outbound Policy Group (safi 7): 0x12db1398 (Hash 0), ID: 2, P#: 0, Drop 0, Use Count: 2,
Staring: 0, Update: 0, CCnt: 0
    Ribout Group: 0x2081a870, ID: 2, Type: 1, Peer Count: 2, Mask: 0x00000003 (1), ribout: 947,
withdrawn: 0
    Stale Route cal info : 0.0.0.0/0, next 0, schd 0
...

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv4 network

Displays BGP4-related debug information for the specified network.

## Syntax

```
show system internal bgp ipv4 network ip-address/mask [ vrf name ]
```

## Parameters

*ip-address/mask*

Specifies an IP address and mask.

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.



## Examples

This example shows sample output from the **show system internal bgp ipv4 network** command.

```
device# show system internal bgp ipv4 network 3.0.0.3/32

*** show ip bgp debug network A.B.C.D A.B.C.D ***
BGP: network 3.0.0.3/32 not found
*** show ip bgp debug route A.B.C.D A.B.C.D ****
tree_node:0x12e6a440, nlri#:1, fwd_route:0x00000000, time:000153ab
color 1, internal_ibgp_route 0, safi 0 , number_of_nlri_rejected 0
installOnly 2,needToAdvertise 2,deleteRibOut 2,rtmWaitingFlag 0,previouslyAdded 0, inChangeList 0
num_ribAdd 0,num_ribAdd_ack 0 ,num_ribAdd_Nack 0 , last_forwarding_route_modify_time_ack:00000000

NLRI:3.0.0.3/32 Peer#: (3) 3.0.0.3 state=6, address:0x12e7d414
tag:00000000
route_is B:1 M:0 L:0 installed:0 inactive:1 filtered:0, rm:0
route_is Is VPN:0 Rt_src:0 Rt_Type:0 exported:0
admin_distance:20 from_IBGP:0 from_Conf_EBGP:0 age:000153ab
route_is aggregate:0 suppressed:0 summary:0 Tree_node:0x12e6a440
As_path:0x12e9fcf7 next_nlri:0x12e7d194, prev_nlri:0x12e7d514, magic 786
----- AS PATH Entry start -----
1      Next Hop   : 3.0.0.3                MED      :0                Origin:INCOMP
      Originator:0.0.0.0                Cluster List:None
      Aggregator:AS Number :0                Router-ID:0.0.0.0                Atomic:None
      Local Pref:100                    Communities:Internet
      AS Path    :2 (length 1)
          AsPathLen: 1  AsNum: 1, SegmentNum: 1, Neighboring As: 2, Source As 2
      AsPath_Addr: 0x12e9fcf7 Nh_Addr: 0x12e329a6 Nlri_Addr: 0x12e7d414 Hash:1913 (0x0300010a)
      Links: 0x00000000, 0x00000000
      Reference Counts: 1:0:1, Magic: 786
----- AS PATH Entry ends -----
as_path segments:0x12e88076 #AS:1(0), #Seq:1(0), Len:1
communities:0x12e9fd3b called:0, check_filter:0, no_export:0, no_advertise:0, local_as:0
neighboring as: 2, source as: 2, next-hop: 3.0.0.3, orig_as_path:0x00000000
----- Nexthop address start -----
Next-hop 3.0.0.3, safi 0, changed 0, used 0, igp_route_type 1, igp_route_sub_type 2, igp_route_cost
0, plen 32, number_of_paths 1
  last_update_time 86954 , resolve_source 0 , reCalculate 0
  ribSubscribeState 3, ribSubScribe_time 000153aa, ribResolved_time 000153aa
  i 0, next_hop_router_ip_address 0.0.0.0, outgoing_interface po39(166)
----- Nexthop address ends -----
same bitLen next:0x00000000, prev:0x00000000
aggregate:0x00000000, damping_nlri:00000000
RIB_out:0x12e5a72c, #_RIB_out:1, weight:0
route_is_restart_stale 0,route_is_special_network_local 0
waiting_for_label 0 , ospf_ext_type2 0 , ospf_rt_type 0 , route_is_new 0 , route_is_valid nexthop 1
last_forwarding_route_modify_time 00000000, last_forwarding_route_modify_time_ack 00000000
Tx Path Id 1, Rx Path Id 0
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv4 nexthop

Displays debug information related to BGP4 next hops in the system.

## Syntax

```
show system internal bgp ipv4 nexthop [ vrf name ]
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv4 nexthop** command.

```
device# show system internal bgp ipv4 nexthop

Next-hop 0.0.0.0 Nh_Addr: 0x12e3173f Aspath_Addr : 0x12e24388
  safi 0, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 0 , afi
1 , bgp_initiated_tnl 0
  ribSubscribeState 1 , ribSubscribe_time 0000052c, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 3.0.0.3 Nh_Addr: 0x12e329a6 Aspath_Addr : 0x12e9fc7c
  safi 0, changed 0, used 0, igp_route_type 1, igp_route_sub_type 2, igp_route_cost 0, plen 32,
number_of_paths 1,
  vr_ribVrfIdx 1 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 0 , afi
1 , bgp_initiated_tnl 0
  ribSubscribeState 3 , ribSubscribe_time 000153aa, ribResolved_time 000153aa
  last_update_time 86954 , resolve_source 0 , reCalculate 0
  i 0, next_hop_router_ip_address 0.0.0.0, outgoing_interface 166
  nexthop_cache_miss_count 3
  nexthop_cache_hit_count 0

Next-hop 0.0.0.0 Nh_Addr: 0x12e30f5c Aspath_Addr : 0x12e207f4
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 7 , afi
1 , bgp_initiated_tnl 0
  ribSubscribeState 0 , ribSubscribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 6.0.0.6 Nh_Addr: 0x12e32ee8 Aspath_Addr : 0x12e9fd72
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 1 , safi 7 , afi
1 , bgp_initiated_tnl 1
  ribSubscribeState 0 , ribSubscribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 9.0.0.9 Nh_Addr: 0x12e311fd Aspath_Addr : 0x12e21c22
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 7 , afi
1 , bgp_initiated_tnl 0
  ribSubscribeState 0 , ribSubscribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 78.0.0.78 Nh_Addr: 0x12e3342a Aspath_Addr : 0x12e28747
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 1 , safi 7 , afi
1 , bgp_initiated_tnl 1
  ribSubscribeState 0 , ribSubscribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0
  nexthop_cache_miss_count 6
  nexthop_cache_hit_count 0
  Total number of BGP Attribute Entries: 550
```

show system internal bgp ipv4 nexthop

```
1      Next Hop   : 78.0.0.78           MED       :0           Origin:INCOMP
      Originator:0.0.0.0           Cluster List:None
      Aggregator:AS Number :0       Router-ID:0.0.0.0           Atomic:None
      Local Pref:100               Communities:Internet
      Extended Community: ExtCom:03:0d:00:00:00:00:00:00 ExtCom:03:0c:00:00:00:00:00:08 RT 3:11 RT

1:1 RT 3:20 ExtCom:06:03:50:eb:1a:13:ce:f5
      AS Path      :2 3 (length 3)
      AsPathLen: 2  AsNum: 2, SegmentNum: 1, Neighboring As: 2, Source As 3
      AsPath_Addr: 0x12e27580 Nh_Addr: 0x12e3342a Nlri_Addr: 0x2628d4c4 Hash:49 (0x03300023a)
      Links: 0x00000000, 0x00000000
      Reference Counts: 4:0:2, Magic: 1254
...

```

## Examples

This example shows sample output from the **show system internal bgp ipv4 nexthop** command for VRF "blue".

```
device# show system internal bgp ipv4 nexthop vrf blue

Next-hop 0.0.0.0 Nh_Addr: 0x12e31f22 Aspath_Addr : 0x12e24d9f
 safi 0, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
 number_of_paths 0,
 vr_ribVrfIdx 0 , vr_vrfIdx 2 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 0 , afi
 1 , bgp_initiated_tnl 0
 ribSubscribeState 1 , ribSubScribe_time 000009b1, ribResolved_time 00000000
 last_update_time 0 , resolve_source 0 , reCalculate 0
 nexthop_cache_miss_count 1
 nexthop_cache_hit_count 0

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv4 tcpdump

Displays debug information related to BGP4 TCP connections.

## Syntax

```
show system internal bgp ipv4 tcpdump [ vrf name ]
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv4 tcpdump** command.

```
device# show system internal bgp ipv4 tcpdump

Key (sockfd) :000 , key_len :29

Key (sockfd) :00075 , key_len :32
  VRF Index: 1
  LocalIP: 1000:3:9::1, Port: 179
  RemoteIP: 1000:3:9::, Port: 8052
  >> Peer_number index 1, handle 75 , Outgoing port_number 166 LocalIP :1000:3:9:: RemoteIP
:1000:3:9:: (1000:3:9::)

Key (sockfd) :00076 , key_len :32
  VRF Index: 1
  LocalIP: 9.0.0.9, Port: 179
  RemoteIP: 3.0.0.3, Port: 8060
  >> Peer_number index 3, handle 76 , Outgoing port_number 14564 LocalIP :3.0.0.3 RemoteIP :3.0.0.3
(3.0.0.3)

pollfd index:1 fd:75 events:8193 revents:0
  getsockopt ret:0 SO_NSID(42) vrf_index:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_KEEPALIVE keepalive:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_DEBUG Debug:0 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPCNT keepalive_count:2 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPINTVL keepalive_int:60 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPIDLE keepalive_Idle:120 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ttl:64 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ip_minttl:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_TCLASS ipiv6_tclass:192 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_UNICAST_HOPS ipiv6_hoplimit:64 optlen:4
  mirror_poll_fd index:1 fd:75 events:8193 revents:0

pollfd index:3 fd:76 events:8193 revents:0
  getsockopt ret:0 SO_NSID(42) vrf_index:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_KEEPALIVE keepalive:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_DEBUG Debug:0 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPCNT keepalive_count:2 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPINTVL keepalive_int:60 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPIDLE keepalive_Idle:120 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ttl:5 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ip_minttl:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_TCLASS ipiv6_tclass:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_UNICAST_HOPS ipiv6_hoplimit:64 optlen:4
  mirror_poll_fd index:3 fd:76 events:8193 revents:0
...
```

## Examples

This example shows sample output from the **show system internal bgp ipv4 tcpdump** command for VRF "red".

```
device# show system internal bgp ipv4 tcpdump vrf red

Key (sockfd) :000 , key_len :29

Key (sockfd) :00075 , key_len :32
  VRF Index: 1
  LocalIP: 1000:3:9::1, Port: 179
  RemoteIP: 1000:3:9::, Port: 8052
  >> Peer_number index 1, handle 75 , Outgoing port_number 166 LocalIP :1000:3:9:: RemoteIP :1000:3:9::
  (1000:3:9::)

Key (sockfd) :00076 , key_len :32
  VRF Index: 1
  LocalIP: 9.0.0.9, Port: 179
  RemoteIP: 3.0.0.3, Port: 8060
  >> Peer_number index 3, handle 76 , Outgoing port_number 14564 LocalIP :3.0.0.3 RemoteIP :3.0.0.3
  (3.0.0.3)

pollfd index:1 fd:75 events:8193 revents:0
  getsockopt ret:0 SO_NSID(42) vrf_index:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_KEEPALIVE keepalive:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_DEBUG Debug:0 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPCNT keepalive_count:2 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPINTVL keepalive_int:60 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPIDLE keepalive_Idle:120 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ttl:64 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ip_minttl:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_TCLASS ipiv6_tclass:192 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_UNICAST_HOPS ipiv6_hoplimit:64 optlen:4
  mirror_poll_fd index:1 fd:75 events:8193 revents:0

pollfd index:3 fd:76 events:8193 revents:0
  getsockopt ret:0 SO_NSID(42) vrf_index:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_KEEPALIVE keepalive:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_DEBUG Debug:0 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPCNT keepalive_count:2 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPINTVL keepalive_int:60 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPIDLE keepalive_Idle:120 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ttl:5 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ip_minttl:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_TCLASS ipiv6_tclass:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_UNICAST_HOPS ipiv6_hoplimit:64 optlen:4
  mirror_poll_fd index:3 fd:76 events:8193 revents:0
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv4 variables

Displays debug information about the BGP4 internal system variables for internal use.

## Syntax

```
show system internal bgp ipv4 variables [ vrf name ]
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.



## Examples

This example shows sample output from the **show system internal bgp ipv4 variables** command.

```
device# show system internal bgp ipv4 variables

*** show ip bgp debug variables safi: 0 ***
safi:0, &bgp:3fd354a8, enabled:1, operational:1, trace_dbg_mem =0, curr_afi:1 vr_init:1
io_process_running:0, io_process_next_peer_number=0
in_long_loops 0, clear_all 0, timer 00000000, count 0
timer_enabled:1, timer_next_peer_number:0, 1s timer 1, short timer 1
scheduler id:5:9, ip:0.0.0.0/0, time=2907311
bgp_tcb:10760d40 (0x00000000, 67), tick_cnt=0, seconds=15
bgp_tcb6:3fd349b0 (0x00000000, 0x00000043)
Auto shutdown new neighbor enabled= 0
*peer:1076293c, *peer_group:10786af0, RIB_in_root_node:12e6a310
Maximum Peer Index Number:4, check_nexthops:0 0
router id:9.0.0.9, configured:1, cluster_id:0.0.0.0, configured:0
route_is_router_reflector:0, client_to_client_reflection:1
networks:x00000000, aggregate:x00000000
default_metric:0, local_preference:100, keep_alive:60, hold_time:180
originate_default:0, originated:0
vr_originate_default_ribSubscribeState 0, vr_originate_default_ribSubscribe_time 00000000,
vr_originate_default_ribUnSubscribe_time 00000000, vr_originate_default_ribResolved_time 00000000
Rib Route Count (v4:3 v6:6)
Ack_pending (v4:0 v6:0)
distance:20 200 200, fast_external_fallover=0
nexthop recur1, nexthop recur ipv6 1, en_def:0, readvertise:1, auto_sum:0, synch:0
always_compare_med:0, compare_med_with_empty_aspath: 0, redistribute_ibgp:0,

local_network_check_time_count:144930
vr_maximum_runtime_ip_paths[i/e]:8, 8, vr_maximum_runtime_bgp_paths[i/e]:-1, -1
nexthop_cache_hit_count:0, nexthop_cache_miss_count:3
system memory:268848644, available:0; total_allocated:4993082, bgp_defined_quota:500000000
Memory 85Reached: 0, Memory 90Reached 0
v4 import map:"", 0x00000000 v4 export map:"", 0x00000000
v6 import map:"", 0x00000000 v6 export map:"", 0x00000000
v4_nexthop_lb_interface:<no-such-port>, v6_nexthop_lb_interface:<no-such-port>,

v4_nexthop_lb_addr:0.0.0.0
v6_nexthop_lb_addr:0.0.0.0
vrf instances:4
 [safi:0][type:0].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
 [safi:0][type:1].redistribution_enabled 1, ribSubscribeState 1, ribSubscribe_time 0000050a,
ribUnSubscribe_time 00000000, received_routeCount 0
...
```

## Examples

This example shows sample output from the **show system internal bgp ipv4 variables** command for VRF "green".

```
device# show system internal bgp ipv4 variables vrf green

*** show ip bgp debug variables safi(0) vrfId(3) ****
safi:0, &bgp:3fd320b8, enabled:1, operational:1, trace_dbg_mem =0, curr_afi:1 vr_init:1
io_process_running:0, io_process_next_peer_number=0
in_long_loops 0, clear_all 0, timer 00000000, count 0
timer_enabled:1, timer_next_peer_number:0, 1s timer 1, short timer 1
scheduler id:5:200, ip:0.0.0.0/0, time=2824518
bgp_tcb:12db22fc (0x00000000, 70), tick_cnt=0, seconds=8
bgp_tcb6:3fd315c0 (0x00000000, 0x00000046)
Auto shutdown new neighbor enabled= 0
*peer:1076293c, *peer_group:10786af0, RIB_in_root_node:12e6b52c
Maximum Peer Index Number:4, check_nexthops:0 0
router id:0.0.0.0, configured:0, cluster_id:0.0.0.0, configured:0
route_is_router_reflector:0, client_to_client_reflection:1
networks:x00000000, aggregate:x00000000
default_metric:0, local_preference:100, keep_alive:60, hold_time:180
originate_default:0, originated:0
vr_originate_default_ribSubscribeState 0, vr_originate_default_ribSubscribe_time 00000000,
vr_originate_default_ribUnSubscribe_time 00000000, vr_originate_default_ribResolved_time 00000000
Rib Route Count (v4:0 v6:0)
Ack_pending (v4:0 v6:0)
distance:20 200 200, fast_external_fallover=0
nexthop recur0, nexthop recur ipv6 0, en_def:0, readvertise:1, auto_sum:0, synch:0
always_compare_med:0, compare_med_with_empty_aspath: 0, redistribute_ibgp:0,
local_network_check_time_count:145100
vr_maximum_runtime_ip_paths[i/e]:8, 8, vr_maximum_runtime_bgp_paths[i/e]:-1, -1
nexthop_cache_hit_count:0, nexthop_cache_miss_count:4
system memory:268848644, available:0; total_allocated:4993082, bgp_defined_quota:500000000
Memory 85Reached: 0, Memory 90Reached 0
v4 import map:"", 0x00000000 v4 export map:"", 0x00000000
v6 import map:"", 0x00000000 v6 export map:"", 0x00000000
v4_nexthop_lb_interface:<no-such-port>, v6_nexthop_lb_interface:<no-such-port>, v4_nexthop_lb_addr:
0.0.0.0
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv6 neighbor

Displays BGP4+-related debug info for the specified neighbor.

## Syntax

```
show system internal bgp ipv6 network ipv6-address [ vrf name ]
```

## Parameters

*ipv6-address*

Specifies an IPv6 address.

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv6 neighbor** command.

```
device# show system internal bgp ipv6 neighbor 1000:2:9::

index:0, base Address:12d99c78
IP:1000:2:9:: (1000:2:9::) Remote_AS:2 (2) Enable:1 Shutdown:0 (generate-rib-out 0), 0
seconds_between_connection_attempts:10 current_seconds:10, up:0
LocalIP:0.0.0.0 Hold_timer:180 KeepAlive:60 Configured=0
safi: nego 0, adv 0, rcv 0, send_disable 0, enabled 1 (1)
GR: send_enable 0, sent 0, R_bit 0, this safi 0, Fbit 0, eor_s 0
    rcv restart 0, R_bit 0, restart_time 0, this safi 0, Fbit 0, eor_w 0
        stale_state 0, restarting_state 0
LAST: sent 0, R_bit 0, this safi 0, Fbit 0
    rcv restart 0, R_bit 0, restart_time 0, this safi 0, Fbit 0
GR Timers : purge_time_count 0, restart_time_count 0, staleroute_time_count 0 reStartStaleTimer 0
Fill Index: 0, Detach Index: 0
Weight:0 Local_Pref:100
Max_Pref=0 Threshold_Percent:0 Teardown:0
Warning_Limit=0 Temporary_Shutdown:0 Exceed_Warning_Limit:0
ebgp_multihop:0 ttl:0, btsh: 0
maxas-limit:300
enforce-first-as:0
Password Encrypt Code:0, String:<>
Password length:0, String:<xxxxxx> Next_hop_self:0, always 0
SendCommunity:0, Send-label: 0, Remove_Private_as:0 Reflector_Client:0 Static_network_edge: 0
As_Override:0, Allowas_in count=5
Orig_Def:0, Orig_Def_Route-Map:<> 00000000
Default tx count:0 - 0, (0) rx count: 0 - 0
After changing additional-paths is clear pending: No
Add Path Capability enabled for safi: No
Add Path Capability Negotiation for sending: FALSE for receiving: FALSE
Route advertisement interval 0 seconds, last send at 146831 seconds ago, Peer ready to send updates
AS4 capability negotiated = 0
AS4 capability configured = 0
AsPathFilter In:0, Number:0 Out:0 Number:0
AddrFilter In:0, Number:0 Out:0 Number:0
PrefixList In:<>, 00000000 Out:<>, 00000000
AsPathACL In:, Out: CommunityACL In:, Out:
AddressPrefixOptionReceived: 0, AddressPrefixConfigOption: 0, AddressPrefixOptionNego: 0, orf
00000000
AllowedToSendUpdate: 0, SendPrefixOrfUpdate: 0, PrefixOrfSentCount: 0
ExtCommOrfOptionReceived: 0, ExtCommOrfConfigOption: 0, ExtCommOrfOptionNego: 0
SendExtcommOrfUpdate: 0, ExtcommOrfSentCount: 0
ExtComm Imported ORFs: []
IpACL In:0 Out:0
IpACLName In:<>, 00000000 Out:<>, 00000000
Current Remote_as:2 State:2 clearing safi 0, 0, 0, 0, 0, '*' 0/0/0, '+' 0
Tcp_close_pending :0
    Outbound Policy Group (safi 2): 0x12e1ff98 (Hash 0), ID: 2, P#: 0, Drop 0,
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv6 network

Displays BGP4+-related debug information for the specified network.

## Syntax

```
show system internal bgp ipv6 network { ipv6-address | ipv6-address/mask } [ vrf name ]
```

## Parameters

*ipv6-address*

Specifies an IPv6 address.

*ipv6-address/mask*

Specifies an IPv6 address and mask.

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv6 network** command.

```

device# show system internal bgp ipv6 network 1000:3::3/128
*** show ip bgp debugp network X:X::X:X <1-128> ***
BGP: network 1000:3::3/128 not found
*** show ip bgp debug route X:X::X:X <1-128> ***
tree_node:0x12e6a35c, nlri#:1, fwd_route:0x00000000, time:00000acf
color 0, internal_ibgp_route 0, safi 2 , number_of_nlri_rejected 0
installOnly 2,needToAdvertise 2,deleteRibOut 2,rtmWaitingFlag 0,previouslyAdded 1, inChangeList 0
num_ribAdd 1,num_ribAdd_ack 1 , num_ribAdd_Nack 0 , last_forwarding_route_modify_time_ack:00000acf

NLRI:1000:3::3/128 Peer#:(1) 1000:3:9:: state=6, address:0x12e7d214
tag:00000000
route_is B:1 M:0 L:0 installed:1 inactive:0 filtered:0, rm:0
route_is Is_VPN:0 Rt_src:0 Rt_Type:0 exported:0
admin_distance:20 from_IBGP:0 from_Conf_EBGP:0 age:00000acf
route_is aggregate:0 supressed:0 summary:0 Tree_node:0x12e6a35c
As_path:0x12e26c5f next_nlri:0x00000000, prev_nlri:0x12e7d294, magic 210
----- AS PATH Entry start-----
1      Next Hop   : 1000:3:9::                                MED      :0

Origin:INCOMP
  Originator:0.0.0.0          Cluster List:None
  Aggregator:AS Number :0      Router-ID:0.0.0.0      Atomic:None
  Local Pref:100              Communities:Internet
  AS Path :2 (length 1)
    AsPathLen: 1 AsNum: 1, SegmentNum: 1, Neighboring As: 2, Source As 2
  AsPath Addr: 0x12e26c5f Nh Addr: 0x12e32c47 Nlri Addr: 0x12e7d214 Hash:5250 (0x0300010a)
  Links: 0x00000000, 0x00000000
  Reference Counts: 6:0:5, Magic: 210
----- AS PATH Entry ends -----
as_path_segments:0x12e88076 #AS:1(0), #Seq:1(0), Len:1
communities:0x12e26ca3 called:0, check_filter:0, no_export:0, no_advertise:0, local_as:0
neighboring_as: 2, source_as: 2, next-hop: 1000:3:9::, orig_as_path:0x00000000
----- Nexthop address start -----
Next-hop 1000:3:9::, safi 2, changed 0, used 0, igp_route_type 3, igp_route_sub_type 0,

igp_route_cost 0, plen 127, number_of_paths 1
  last_update_time 2767 , resolve_source 0 , reCalculate 0
  ribSubscribeState 3, ribSubScribe_time 00000acf, ribResolved_time 00000acf
  i 0, next_hop_router_ip_address 1000:3:9::, outgoing_interface po39(166)
----- Nexthop address ends -----
same bitLen next:0x00000000, prev:0x00000000
aggregate:0x00000000, damping_nlri:00000000
RIB_out:0x12e5a770, #_RIB_out:1, weight:0
route_is restart stale 0,route_is_special_network_local 0
waiting_for_label 0 , ospf_ext_type2 0 , ospf_rt_type 0 , route_is_new 0 , route_is_valid_nexthop 1
last_forwarding_route_modify_time 00000acf, last_forwarding_route_modify_time_ack 00000acf
Tx Path Id 1, Rx Path Id 0

```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv6 nexthop

Displays debug information related to BGP4+ next hops in the system.

## Syntax

```
show system internal bgp ipv6 nexthop [ vrf name ]
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv6 nexthop** command.

```
device# show system internal bgp ipv6 nexthop

Next-hop :: Nh_Addr: 0x12e3149e Aspath_Addr : 0x12e2430d
  safi 2, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
  number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 2 , afi
  2 ,_bgp_initiated_tnl 0
  ribSubscribeState 1 , ribSubScribe_time 0000052c, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 1000:3:9:: Nh_Addr: 0x12e32c47 Aspath_Addr : 0x12e26c5f
  safi 2, changed 0, used 0, igp_route_type 3, igp_route_sub_type 0, igp_route_cost 0, plen 127,
  number_of_paths 1,
  vr_ribVrfIdx 1 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 2 , afi
  2 ,_bgp_initiated_tnl 0
  ribSubscribeState 3 , ribSubScribe_time 00000acf, ribResolved_time 00000acf
  last_update_time 2767 , resolve_source 0 , reCalculate 0
  i 0, next_hop_router_ip_address 1000:3:9::, outgoing_interface 166
nexthop_cache_miss_count 3
nexthop_cache_hit_count 0

Next-hop 0.0.0.0 Nh_Addr: 0x12e30f5c Aspath_Addr : 0x12e207f4
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
  number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 7 , afi
  1 ,_bgp_initiated_tnl 0
  ribSubscribeState 0 , ribSubScribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 6.0.0.6 Nh_Addr: 0x12e32ee8 Aspath_Addr : 0x12e9fd72
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
  number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 1 , safi 7 , afi
  1 ,_bgp_initiated_tnl 1
  ribSubscribeState 0 , ribSubScribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 9.0.0.9 Nh_Addr: 0x12e311fd Aspath_Addr : 0x12e21c22
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
  number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 7 , afi
  1 ,_bgp_initiated_tnl 0
  ribSubscribeState 0 , ribSubScribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0

Next-hop 78.0.0.78 Nh_Addr: 0x12e3342a Aspath_Addr : 0x12e28747
  safi 7, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
  number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 1 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 1 , safi 7 , afi
  1 ,_bgp_initiated_tnl 1
  ribSubscribeState 0 , ribSubScribe_time 00000000, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0
nexthop_cache_miss_count 6
nexthop_cache_hit_count 0
  Total number of BGP Attribute Entries: 550
1   Next Hop   : 78.0.0.78           MED       :0           Origin:INCOMP
   Originator:0.0.0.0           Cluster List:None
   Aggregator:AS Number :0           Router-ID:0.0.0.0           Atomic:None
   Local Pref:100               Communities:Internet
   Extended Community: ExtCom:03:0d:00:00:00:00:00:00:00:00:00:00:08 RT 3:11 RT
1:1 RT 3:20 ExtCom:06:03:50:eb:1a:13:ce:f5
   AS Path    :2 3 (length 3)
   AsPathLen: 2 AsNum: 2, SegmentNum: 1, Neighboring As: 2, Source As 3
   AsPath_Addr: 0x12e27580 Nh_Addr: 0x12e3342a Nlri_Addr: 0x2628d4c4 Hash:49 (0x0300023a)
   Links: 0x00000000, 0x00000000
   Reference Counts: 4:0:2, Magic: 1254
...
```



## Examples

This example shows sample output from the **show system internal bgp ipv4 nexthop** command for VRF "yellow".

```
device# show system internal bgp ipv6 nexthop vrf yellow

Next-hop :: Nh_Addr: 0x12e31c81 Aspath_Addr : 0x12e24760
  safi 2, changed 0, used 0, igp_route_type 0, igp_route_sub_type 0, igp_route_cost 0, plen 0,
  number_of_paths 0,
  vr_ribVrfIdx 0 , vr_vrfIdx 5 , resolvedRtPlen 0 , resolvedFirsttime 0 , tnl_status 0 , safi 2 , afi
  2 ,_bgp_initiated_tnl 0
  ribSubscribeState 1 , ribSubScribe_time 0000098b, ribResolved_time 00000000
  last_update_time 0 , resolve_source 0 , reCalculate 0
  nexthop_cache_miss_count 1
  nexthop_cache_hit_count 0
LeafA_4# 2016/01/07-16:00:23, [NSM-1002], 9379, SW/0 | Active | DCE, INFO, LeafA_4, Interface
TenGigabitEthernet 109/0/4 is protocol down.
2016/01/07-16:00:23, [NSM-1003], 9380, SW/0 | Active | DCE, INFO, LeafA_4, Interface
TenGigabitEthernet 109/0/4 is link down.
2016/01/07-16:00:25, [NSM-1001], 9381, SW/0 | Active | DCE, INFO, LeafA_4, Interface
TenGigabitEthernet 109/0/4 is online.
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv6 tcpdump

Displays debug information related to BGP4+ TCP connections.

## Syntax

```
show system internal bgp ipv6 tcpdump [ vrf name ]
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv6 tcpdump** command.

```
device# show system internal bgp ipv6 tcpdump

Key (sockfd) :000 , key_len :29

Key (sockfd) :00075 , key_len :32
  VRF Index: 1
  LocalIP: 1000:3:9::1, Port: 179
  RemoteIP: 1000:3:9::, Port: 8052
  >> Peer_number index 1, handle 75 , Outgoing port_number 166 LocalIP :1000:3:9:: RemoteIP
:1000:3:9:: (1000:3:9::)

Key (sockfd) :00076 , key_len :32
  VRF Index: 1
  LocalIP: 9.0.0.9, Port: 179
  RemoteIP: 3.0.0.3, Port: 8060
  >> Peer_number index 3, handle 76 , Outgoing port_number 14564 LocalIP :3.0.0.3 RemoteIP :3.0.0.3
(3.0.0.3)

pollfd index:1 fd:75 events:8193 revents:0
  getsockopt ret:0 SO_NSID(42) vrf_index:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_KEEPALIVE keepalive:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_DEBUG Debug:0 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPCNT keepalive_count:2 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPINTVL keepalive_int:60 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPIDLE keepalive_Idle:120 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ttl:64 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ip_minttl:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_TCLASS ipiv6_tclass:192 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_UNICAST_HOPS ipiv6_hoplimit:64 optlen:4
  mirror_poll_fd index:1 fd:75 events:8193 revents:0

pollfd index:3 fd:76 events:8193 revents:0
  getsockopt ret:0 SO_NSID(42) vrf_index:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_KEEPALIVE keepalive:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_DEBUG Debug:0 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPCNT keepalive_count:2 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPINTVL keepalive_int:60 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPIDLE keepalive_Idle:120 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ttl:5 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ip_minttl:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_TCLASS ipiv6_tclass:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_UNICAST_HOPS ipiv6_hoplimit:64 optlen:4
  mirror_poll_fd index:3 fd:76 events:8193 revents:0...
```

## Examples

This example shows sample output from the **show system internal bgp ipv4 tcpdump** command for VRF "yellow".

```
device# show system internal bgp ipv6 tcpdump vrf yellow

Key (sockfd) :000 , key_len :29

Key (sockfd) :00075 , key_len :32
  VRF Index: 1
  LocalIP: 1000:3:9::1, Port: 179
  RemoteIP: 1000:3:9::, Port: 8052
  >> Peer_number index 1, handle 75 , Outgoing port_number 166 LocalIP :1000:3:9:: RemoteIP :1000:3:9::
  (1000:3:9::)

Key (sockfd) :00076 , key_len :32
  VRF Index: 1
  LocalIP: 9.0.0.9, Port: 179
  RemoteIP: 3.0.0.3, Port: 8060
  >> Peer_number index 3, handle 76 , Outgoing port_number 14564 LocalIP :3.0.0.3 RemoteIP :3.0.0.3
  (3.0.0.3)

pollfd index:1 fd:75 events:8193 revents:0
  getsockopt ret:0 SO_NSID(42) vrf_index:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_KEEPALIVE keepalive:1 optlen:4
  getsockopt ret:0 SOL_SOCKET/SO_DEBUG Debug:0 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPCNT keepalive_count:2 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPINTVL keepalive_int:60 optlen:4
  getsockopt ret:0 SOL_TCP/TCP_KEEPIDLE keepalive_Idle:120 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ttl:64 optlen:4
  getsockopt ret:0 IPPROTO_IP/IP_TTL ip_minttl:0 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_TCLASS ipv6_tclass:192 optlen:4
  getsockopt ret:0 IPPROTO_IPV6/IPV6_UNICAST_HOPS ipv6_hoplimit:64 optlen:4
mirror_poll_fd index:1 fd:75 events:8193 revents:0
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal bgp ipv6 variables

Displays debug information about the BGP4+ internal system variables for internal use.

## Syntax

```
show system internal bgp ipv6 variables [ vrf name ]
```

## Parameters

*vrf name*

Specifies the name of the VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

This example shows sample output from the **show system internal bgp ipv6 variables** command.

```

device# show system internal bgp ipv6 variables
*** show ip bgp debug variables safi: 2 ****
safi:2, &bgp:3fd354a8, enabled:1, operational:1, trace_dbg_mem =0, curr_afi:2 vr_init:1
io_process_running:0, io_process_next_peer number=0
in_long_loops 0, clear_all 0, timer 00000000, count 0
timer_enabled:1, timer_next_peer number:0, ls timer 1, short timer 1
scheduler id:5:9, ip::/0, time=2918651
bgp_tcb:10760d40 (0x00000000, 67), tick_cnt=0, seconds=10
bgp_tcb6:3fd349b0 (0x00000000, 0x00000043)
Auto shutdown new neighbor enabled= 0
*peer:1076293c, *peer_group:10786af0, RIB_in_root_node:12e6a2c4
Maximum Peer Index Number:4, check_nexthops:0 0
router_id:9.0.0.9, configured:1, cluster_id:0.0.0.0, configured:0
route_is_router_reflector:0, client_to_client_reflection:1
networks:x00000000, aggregate:x00000000
default_metric:0, local_preference:100, keep_alive:60, hold_time:180
originate_default:0, originated:0
vr_originate_default_ribSubscribeState 0, vr_originate_default_ribSubscribe_time 00000000,
vr_originate_default_ribUnSubscribe_time 00000000, vr_originate_default_ribResolved_time 00000000
Rib Route Count (v4:4 v6:8)
Ack_pending (v4:0 v6:0)
distance:20 200 200, fast_external_fallover=0
nexthop recur1, nexthop recur ipv6 1, en_def:0, readvertise:1, auto_sum:0, synch:0
always_compare_med:0, compare_med_with_empty_aspath: 0, redistribute_ibgp:0,
local_network_check_time_count:145497
vr_maximum_runtime_ip_paths[i/e]:8, 8, vr_maximum_runtime_bgp_paths[i/e]:-1, -1
nexthop_cache_hit_count:0, nexthop_cache_miss_count:3
system memory:268848644, available:0; total_allocated:5021521, bgp_defined_quota:500000000
Memory 85Reached: 0, Memory 90Reached 0
v4 import map:"", 0x00000000 v4 export map:"", 0x00000000
v6 import map:"", 0x00000000 v6 export map:"", 0x00000000
v4_nexthop_lb_interface:<no-such-port>, v6_nexthop_lb_interface:<no-such-port>,
v4_nexthop_lb_addr:0.0.0.0
v6_nexthop_lb_addr:0.0.0.0
vrf instances:4
[safi:2][type:0].redistribution_enabled 0, ribSubscribeState 0, ribSubscribe_time 00000000,
ribUnSubscribe_time 00000000, received_routeCount 0
[safi:2][type:1].redistribution_enabled 1, ribSubscribeState 1, ribSubscribe_time 00000506,
ribUnSubscribe_time 00000000, received_routeCount 0
...

```

## Examples

This example shows sample output from the **show system internal bgp ipv6 variables** command for VRF "yellow".

```
device# show system internal bgp ipv6 variables vrf yellow

safi:2, &bgp:3fd320b8, enabled:1, operational:1, trace_dbg_mem =0, curr_afi:2 vr_init:1
io_process_running:0, io_process_next_peer number=0
in_long_loops 0, clear_all 0, timer 00000000, count 0
timer_enabled:1, timer_next_peer_number:0, ls timer 1, short timer 1
scheduler id:5:200, ip::/0, time=2834944
bgp_tcb:12db391c (0x00000000, 72), tick_cnt=0, seconds=12
bgp_tcb6:3fd315c0 (0x00000000, 0x00000048)
Auto shutdown new neighbor enabled= 0
*peer:1076293c, *peer_group:10786af0, RIB_in_root_node:12e69698
Maximum Peer Index Number:4, check_nexthops:0 0
router_id:0.0.0.0, configured:0, cluster_id:0.0.0.0, configured:0
route_is_router_reflector:0, client_to_client_reflection:1
networks:x00000000, aggregate:x00000000
default_metric:0, local_preference:100, keep_alive:60, hold_time:180
originate_default:0, originated:0
vr_originate_default_ribSubscribeState 0, vr_originate_default_ribSubscribe_time 00000000,
vr_originate_default_ribUnSubscribe_time 00000000, vr_originate_default_ribResolved_time 00000000
Rib Route Count (v4:0 v6:0)
Ack_pending (v4:0 v6:0)
distance:20 200 200, fast_external_fallover=0
nexthop recur0, nexthop recur ipv6 0, en_def:0, readvertise:1, auto_sum:0, synch:0
always_compare_med:0, compare_med_with_empty_aspath: 0, redistribute_ibgp:0,
local_network_check_time_count:145621
vr_maximum_runtime_ip_paths[i/e]:1, 1, vr_maximum_runtime_bgp_paths[i/e]:1, 1
nexthop_cache_hit_count:0, nexthop_cache_miss_count:1
system memory:1070801088, available:271338324, total_allocated:5021521, bgp_defined_quota:500000000
Memory 85Reached: 0, Memory 90Reached 0
v4 import map:"", 0x00000000 v4 export map:"", 0x00000000
v6 import map:"", 0x00000000 v6 export map:"", 0x00000000
...
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show system internal dcm

Displays distributed configuration management (DCM) information in the system.

## Syntax

```
show system internal dcm { clients | last-config-time xpaths | memstat [ detail ] | message-stat all | object-stat all }
```

```
show system internal dcm message details config service service-number { off | on }
```

```
show system internal dcm service [ details ] service-number
```

```
show system internal dcm vlan { port-vlans | provisioned-vlans | vlans-with-ivid }
```

## Parameters

### clients

Displays connected clients.

### last-config-time xpaths

Displays last configuration-time xpaths.

### memstat

Displays DCM memory statistics.

#### detail

Displays detailed DCM memory statistics.

### message-stat all

#### ATTENTION

Running this command can use significant system resources.

Displays a summary of all DCM messages.

### object-stat all

Displays a summary of DCM object statuses.

### message details config service *service-number*

Turns on and off a DCM message-history dump for a specified service number. The default is **off**.

#### off

Turns off the specified message-history dump.

#### on

Turns on the specified message-history dump.

### service

Displays detailed or summary information for a DCM service.

#### details

Displays detailed information. If this option is not specified, displays summary information.

#### *service-number*

Specifies a service number.



**vlan**

Displays VLAN-related details.

**port-vlans**

Displays port-VLAN associations.

**provisioned-vlans**

Displays provisioned VLANs.

**vlans-with-ivid**

Displays VLANs associated with IVID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following command displays DCM connected clients.

```
device# show system internal dcm clients
Client Name
-----
DPoD_License0
IGMP0
LldpDCMClient0
RAS0
Sflagt_p00
Sflow0
UlldDCMClient0
WaveClient010.25.224.179810
agd0
arp0
bfd0
bfd0
bgp0
dauthd0
eld0
essd0
fabricd0
fcoed0
fibagt_p00
iphelptd0
l2agt_p00
l2sys0
lacp0
mc_agt.0.0
mcast_ss0
mdd0
mld0
mstp0
nsm0
ofagt_p00
openflow0
ospf0
ospf60
pcap0
pem0
pim0
qos0
radv0
rmon0
rps0
rtm0
snmp0
srm0
ssm0
tnl0
tnlagt_p00
toam0
vrrp0
```

The following command displays last-configuration-time xpaths.

```
device# show system internal dcm last-config-time xpaths
1406825931 [/]
1402732955 [/cee-map]
1403773759 [/fcoe/fcoe-fabric-map]
1402732955 [/hardware/flexport/flexport_type]
1402732955 [/interface-vlan]
1402732955 [/interface-vlan/interface/ve]
1402732955 [/interface-vlan/interface/vlan]
1403780135 [/interface/fc-port]
1406246891 [/interface/fortygigabitethernet]
1402732955 [/interface/fortygigabitethernet/channel-group]
1402732955 [/interface/fortygigabitethernet/switchport]
1402732955 [/interface/fortygigabitethernet/vlan]
1402732955 [/interface/gigabitethernet]
1402732955 [/interface/gigabitethernet/channel-group]
1402732955 [/interface/gigabitethernet/switchport]
1402732955 [/interface/gigabitethernet/vlan]
1402732955 [/interface/hundredgigabitethernet]
1402732955 [/interface/hundredgigabitethernet/channel-group]
1402732955 [/interface/hundredgigabitethernet/switchport]
1402732955 [/interface/hundredgigabitethernet/vlan]
1402732955 [/interface/port-channel]
1402732955 [/interface/port-channel/switchport]
1402732955 [/interface/port-channel/vlan]
1406623387 [/interface/tengigabitethernet]
1402732955 [/interface/tengigabitethernet/channel-group]
1402732955 [/interface/tengigabitethernet/switchport]
1402732955 [/interface/tengigabitethernet/vlan]
1403869536 [/mac-group]
1403860454 [/mac/access-list]
1404472504 [/port-profile]
1402732955 [/port-profile-global/port-profile]
1404549043 [/protocol/spanning-tree]
1403352542 [/rbridge-id/interface/ve]
1404560420 [/snmp-server/agtconfig]
```

SUCCESS: Mon Aug 4 15:41:32 2014 : Success in getLastUpdateTimestampsForXPathStrings operation.

The following command displays a summary of DCM object statuses.

```
device# show system internal dcm object-stat all
INFO : Mon Aug 4 17:08:56 2014 : Leaked Object Summary For Service : 'Wave Database'
SUCCESS: Mon Aug 4 17:08:56 2014 : NO Object Leaks Found.
INFO : Mon Aug 4 17:08:56 2014 : Leaked Object Summary For Service : 'Persistence Local Object
Manager'
SUCCESS: Mon Aug 4 17:08:56 2014 : NO Object Leaks Found.
INFO : Mon Aug 4 17:08:56 2014 : Leaked Object Summary For Service : 'Persistence'
ERROR : Mon Aug 4 17:08:56 2014 : 0000001 ConfigurationInfoManagedObject 161484252:1103806595073
TRUE
ERROR : Mon Aug 4 17:08:56 2014 : 0000002 ConfigurationInfoManagedObject 161484252:1103806595076
TRUE
ERROR : Mon Aug 4 17:08:56 2014 : 0000003 ConfigurationInfoManagedObject 161484252:1103806595075
TRUE
ERROR : Mon Aug 4 17:08:56 2014 : 0000004 ConfigurationInfoManagedObject 161484252:1103806595074
TRUE
```

<Output truncated>

# show system internal nas

Displays all network-attached storage (NAS) server IP addresses and NAS configuration details in the system.

## Syntax

`show system internal nas`

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following example displays NAS configuration information and IP addresses for all active NAS servers.

```
device# show system internal nas
Rbridge 1
-----
Auto-NAS Enabled
Cos 2
Dscp 10
Traffic Class 5
nas server-ip 10.192.100.100/32 vlan 100
nas server-ip 10.192.100.101/32 vrf brown
```

# show system internal nsm

Displays network service module (NSM) information in the system.

## Syntax

```
show system internal nsm { gvlan [ vlan-id ] | ivid [ vlan-id ] | vrbid }
```

## Parameters

### gvlan

Displays global-VLAN (GVLAN) information.

*vlan-id*

Displays GVLAN information for a specified VLAN.

### ivid

Displays information for VLANs associated with IVIDs.

*vlan-id*

Displays GVLAN information for a specified VLAN.

### vrbid

Displays virtual RBridge ID information.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following example displays information for IVIDs provisioned to VLAN 1.

```
device# show system internal nsm ivid 1

VID-IVID Mapping      : Uniform

Total # of IVIDs provisioned to Vlans:  5
Total # of free IVIDs :  7145

GVLAN      IVID      # of Ifs
1          1          0
```

The following example is sample output for the **show system internal nsm gvlan** option.

```
device# show system internal nsm gvlan

Vfab enable state : enabled
Vfab en read status, rc : 0 0
Vfab en stage 0, status 0, dis stage 0, status 0

Total # of Vlans configured: 5

Total # of Vlans provisioned: 5

GVLAN      IVID      # of Ifs
1           1         0
2           2         0
1002       1002      0
4093       4093      0
4095       4095      0
```

The following example is sample output for the **show system internal nsm vrbid** option.

```
device# show system internal nsm vrbid

Total vLAGS: 0
Total VRRP_E Sessions: 0
Total Tunnels: 0
=====
Virtual End Point VRB-ID VRB-Type
=====
```

# show system internal nsx

Displays state information related to the NSX controller.

## Syntax

```
show system internal nsx export-vlan-cache
```

```
show system internal nsx { locator-cache | lswitch-cache } [ count ]
```

## Parameters

### export-vlan-cache

Displays the export-VLAN cache.

### locator-cache

Displays the physical-locator cache.

### count

Displays only the number of cache entries.

### lswitch-cache

Displays the logical-switch cache.

### count

Display only the number of cache entries.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following example shows a sample output of the **show system internal nsx export-vlan-cache** option.

```
device# show system internal nsx export-vlan-cache
41-42,1011-3010
(2002 vlans)
```

show system internal nsx

The following example shows a sample output of the **show system internal nsx lswitch-cache** option.

```
device# show system internal nsx lswitch-cache
Logical_Switch UUID          VNI      GVLAN      ObjectId
=====
38a71eb3-8524-3982-af91-29d66ff7f13b  -1        0      262795638:2203318222851
39dbad3a-0765-3c22-a2c7-257ce8e8c71b  555555    41      262795638:2203318222849
c94fbfd9-6515-3faf-b8ac-3608bcabeb75  6000      42      262795638:2203318222850
(3 entries)
```

The following example shows a sample output of the **show system internal nsx locator-cache** option.

```
device# show system internal nsx locator-cache
Physical_Locator UUID      Tun ID      ObjectId
=====
62323fc8-8e7f-367e-9592-99d73f065357  61441      7525106:2203318222849
e08bad93-30f1-35e9-b2c3-808f75e056ce  61442      7525106:2203318222850
(2 entries)
```



# show system internal ofagt

Displays debug information from the openflow agent, per linecard and partition.

## Syntax

```
show system internal ofagt line-card { p0 | p1 } { flow [ all ] | flow_tree | l2fwd_info }
show system internal ofagt line-card { p0 | p1 } chip { all | chip-number [ l2flow_tree | l3flow_tree ] }
show system internal ofagt line-card { p0 | p1 } group { group-number | all }
show system internal ofagt line-card { p0 | p1 } meter { meter-number | all }
show system internal ofagt line-card { p0 | p1 } port { port-number | all }
show system internal ofagt line-card { p0 | p1 } vlan { vlan-id | all }
```

## Parameters

*line-card*

Specifies the linecard number.

**p0**

Specifies partition 0.

**p1**

Specifies partition 1.

**flow**

Specifies flow information.

**all**

Specifies all flow information.

**flow\_tree**

Specifies flow-tree information.

**l2fwd\_info**

Specifies Layer 2 forwarding information.

**chip**

Species chip information.

**all**

Specifies all chip information.

*chip-number*

Specifies a chip.

**l2flow\_tree**

Specifies the Level 2 flow-tree.

**l3flow\_tree**

Specifies the Level 3 flow-tree.

**group**

Specifies group information.

*group-number*  
Specifies a group number.

**all**  
Specifies all groups.

**meter**  
Specifies meter information.

*meter-number*  
Specifies a meter number.

**all**  
Specifies all meters.

**port**  
Specifies port information.

*port-number*  
Specifies a port number.

**all**  
Specifies all ports.

**vlan**  
Specifies VLAN information.

*vlan-id*  
Specifies a VLAN.

**all**  
Specifies all VLANs.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

This command is supported only on the VDX 8770 models.

## Examples

The following example displays openflow information for all chips on partition 0 of linecard 1.

```
device# show system internal ofagt 1 p0 chip all
Chip Id: 14
  local chip id:    5 ifindex: 0x84800280 slot id:    2
Chip Id: 13
  local chip id:    4 ifindex: 0x84800200 slot id:    2
Chip Id: 12
  local chip id:    3 ifindex: 0x84800180 slot id:    2
Chip Id: 11
  local chip id:    2 ifindex: 0x84800100 slot id:    2
Chip Id: 10
  local chip id:    1 ifindex: 0x84800080 slot id:    2
Chip Id: 9
  local chip id:    0 ifindex: 0x84800000 slot id:    2
Chip Id: 5
  local chip id:    5 ifindex: 0x84400280 slot id:    1
--More--          Chip Id: 4
  local chip id:    4 ifindex: 0x84400200 slot id:    1
Chip Id: 3
  local chip id:    3 ifindex: 0x84400180 slot id:    1
Chip Id: 2
  local chip id:    2 ifindex: 0x84400100 slot id:    1
Chip Id: 1
  local chip id:    1 ifindex: 0x84400080 slot id:    1
Chip Id: 0
  local chip id:    0 ifindex: 0x84400000 slot id:    1
```

# show system internal ovsdb

Displays system information from the ovsdb tables.

## Syntax

```
show system internal ovsdb { monitors | schema }
```

```
show system internal ovsdb table name [ count | where column function value ]
```

## Parameters

### monitors

Specifies registered monitors.

### schema

Specifies all ovsdb schemas and tables.

### table *name*

Specifies an ovsdb table.

### count

Specifies the number of rows.

### where

Specifies a condition.

### *column*

Specifies a table column.

### *function*

Specifies a function, for example, =.

### *value*

Specifies the column value.

## Modes

Privileged EXEC mode

## Usage Guidelines

Diagnostic commands are developed and intended for specialized troubleshooting. Please work closely with Extreme Networks technical support in running **debug** or **show system internal** commands and interpreting their results.

## Examples

The following example shows a sample output of the **show system internal ovssdb schema** option.

```
device# show system internal ovssdb schema
Schema: hardware_vtep
Tables:
  Global
  Logical_Binding_Stats
  Logical_Router
  Logical_Switch
  Manager
  Mcast_Macs_Local
  Mcast_Macs_Remote
  Physical_Locator
  Physical_Locator_Set
  Physical_Port
  Physical_Switch
  Tunnel
  Ucast_Macs_Local
  Ucast_Macs_Remote
```

The following example shows a sample output of the **show system internal ovssdb table *name* where** option.

```
device # show system internal ovssdb table Logical_Switch where tunnel_key=444444
===== Row 1 of 1 =====
description      : LS-5000
name             : 6eaf567f-6129-4125-8fa1-d3e5b8cf946c
tunnel_key       : 444444
_uuid           : 39dbad3a-0765-3c22-a2c7-257ce8e8c71c
_version        : 52e15220-eca6-3b7e-bab9-324d2ea7cef5
```

# show system monitor

Displays the overall switch status and the status of the contributors defined as part of the policy.

## Syntax

```
show system monitor [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays the status of the local switch.

```
device# show system monitor

** System Monitor Switch Health Report **
RBridge 128      switch status      : HEALTHY
                 Time of Report      : 2012-06-19 03:18:28
                 Power supplies monitor : HEALTHY
                 Temperatures monitor  : HEALTHY
                 Fans monitor          : HEALTHY
                 CID-Card monitor      : HEALTHY
                 MM monitor            : HEALTHY
                 LC monitor            : HEALTHY
                 SFM monitor           : HEALTHY
                 Flash monitor         : HEALTHY
```

The following example displays the status of all RBridges.

```
device# show system monitor rbridge-id all
** System Monitor Switch Health Report **
RBridge 2      switch status      : MARGINAL
                 Time of Report      : 2014-08-01 17:30:30
                 Power supplies monitor : MARGINAL
                 Temperatures monitor  : HEALTHY
                 Fans monitor          : HEALTHY
                 Flash monitor         : HEALTHY

RBridge 1      switch status      : MARGINAL
                 Time of Report      : 2014-08-01 17:34:27
                 Power supplies monitor : MARGINAL
                 Temperatures monitor  : HEALTHY
                 Fans monitor          : HEALTHY
                 Flash monitor         : HEALTHY
```

## Related Commands

[system-monitor](#), [system-monitor-mail](#)

# show system pstat interface

Displays control-traffic statistics for packets through a specified interface to the CPU.

## Syntax

```
show system pstat interface { <N>gigabitethernet [ rbridge-id / ] slot / port }
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### *rbridge-id*

Specifies an RBridge ID.

### *slot*

Specifies a valid slot number.

### *port*

Specifies a valid port number.

## Modes

Privileged EXEC mode



## Examples

The following example displays output of this command for a ten-gigabit Ethernet interface.

```
device# show system pstat interface TenGigabitEthernet 1/0/3

Total packet received 0
Total vlan tagged packet receive 0
Total packet receive error 0
Total packet transmitted 0
Total packet transmit error 0

stp packet received 0
vlan tagged stp packet receive 0
stp packet receive error 0
stp packet transmitted 0
stp packet transmit error 0

lacp packet received 0
vlan tagged lacp packet receive 0
lacp packet receive error 0
lacp packet transmitted 0
lacp packet transmit error 0

lldp packet received 0
vlan tagged lldp packet receive 0
lldp packet receive error 0
lldp packet transmitted 0
lldp packet transmit error 0

arp packet received 0
vlan tagged arp packet receive 0
arp packet receive error 0
arp packet transmitted 0
arp packet transmit error 0

gmrp packet received 0
vlan tagged gmrp packet receive 0
gmrp packet receive error 0
gmrp packet transmitted 0
gmrp packet transmit error 0

gvrp packet received 0
vlan tagged gvrp packet receive 0
gvrp packet receive error 0
gvrp packet transmitted 0
gvrp packet transmit error 0

eapol packet received 0
vlan tagged eapol packet receive 0
eapol packet receive error 0
eapol packet transmitted 0
eapol packet transmit error 0

cdp packet received 0
vlan tagged cdp packet receive 0
cdp packet receive error 0
cdp packet transmitted 0
cdp packet transmit error 0

igmp packet received 0
vlan tagged igmp packet receive 0
igmp packet receive error 0
igmp packet transmitted 0
igmp packet transmit error 0

ospf packet received 0
vlan tagged ospf packet receive 0
ospf packet receive error 0
ospf packet transmitted 0
ospf packet transmit error 0
```

show system pstat interface

```
vrrp packet received 0
vlan tagged vrrp packet receive 0
vrrp packet receive error 0
vrrp packet transmitted 0
vrrp packet transmit error 0

ospf6 packet received 0
vlan tagged ospf6 packet receive 0
ospf6 packet receive error 0
ospf6 packet transmitted 0
ospf6 packet transmit error 0

vrrp6 packet received 0
vlan tagged vrrp6 packet receive 0
vrrp6 packet receive error 0
vrrp6 packet transmitted 0
vrrp6 packet transmit error 0

icmp6 packet received 0
vlan tagged icmp6 packet receive 0
icmp6 packet receive error 0
icmp6 packet transmitted 0
icmp6 packet transmit error 0

vrrpe packet received 0
vlan tagged vrrpe packet receive 0
vrrpe packet receive error 0
vrrpe packet transmitted 0
vrrpe packet transmit error 0

vrrpe6 packet received 0
vlan tagged vrrpe6 packet receive 0
vrrpe6 packet receive error 0
vrrpe6 packet transmitted 0
vrrpe6 packet transmit error 0

Unknown packet received 0
vlan tagged Unkown packet receive 0
Unknown packet receive error 0
Unknown packet transmitted 0
Unknown packet transmit error 0
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>pstat interface</b> keywords.

# show telnet server status

Displays the current Telnet server status.

## Syntax

```
show telnet server status [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

When Telnet server status is enabled:

```
switch# show telnet server status [rbridge-id  
rbridge-id  
| all]  
Telnet server status: Enabled
```

When Telnet server is disabled in rbridge-id 3:

```
switch# show telnet server status rbridge-id 3  
rbridge-id 3 Telnet server status: Disabled  
switch#
```

## Related Commands

[show running-config telnet server](#), [telnet server shutdown](#)

# show threshold monitor

Displays the current status of environmental thresholds and alerts for interfaces, security, and SFPs.

## Syntax

```
show threshold monitor [ interface all area | security area [ login-violation [ rbridge-id rbridge-id | all ] | telnet-violation
[ rbridge-id rbridge-id | all ] ] | sfp all area [ current | rxp | temperature | txp | voltage ]
```

## Parameters

### interface all area

Displays status of interface thresholds and alerts.

### security area

Displays status of security thresholds and alerts.

### login-violation

Displays status of login violations.

### telnet-violation

Displays status of Telnet violations.

### sfp all area

Displays status of SFP thresholds and alerts.

### current

Amount of current supplied to the SFP transceiver.

### rxp

Amount of incoming laser power, in microWatts ( $\mu$ W).

### temperature

Temperature of the SFP, in degrees Celsius.

### txp

Amount of outgoing laser power, in microWatts ( $\mu$ W).

### voltage

Amount of voltage supplied to the SFP.

### rbridge-id *rbridge-id*

Specifies a switch by means of the switch's RBridge ID.

### all

Reports status for all nodes in the cluster.

## Modes

Privileged EXEC mode

## Examples

```
switch# show threshold monitor security area login-violation rbridge-id all
Rbridge-Id   Area                Value  Status  Monitoring Status
154          Login Violation      0      In Range Monitoring
```

## Related Commands

[system-monitor](#), [system-monitor-mail](#), [threshold-monitor interface](#)

# show track summary

Displays link-state tracking (LST) details for interfaces on a device.

## Syntax

```
show track summary [ rbridge-id { rbridge-id | range | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge, a set of RBridges, or all RBridges.

#### *rbridge-id*

Specifies an RBridge. The range of valid values is from 1 through 239.

#### *range*

Specifies a range of RBridges. The range string can be discontinuous, such as "1-3,5".

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Command Output

The **show track summary** command displays the following information:

Output field	Description
Tracking (downstream) Interfaces	Displays the downstream (tracking) interfaces.
Tracked (upstream) Interfaces	Displays the upstream (tracked) interfaces.
Admin State	Indicates if LST is enabled or disabled on the interface.
Minlink	Upon failure of one or more uplinks, specifies the minimum number of functioning uplinks below which LST shuts down all downlinks.
Up	The interface is up.
Dwn	The interface is down.
Dis	The interface is disabled.

## Examples

The following example displays the **show track summary** command for a switch on which LST is enabled on one downlink and disabled on one downlink. Min-link is defined.

```
device# show track summary

(Up): Downstream interface admin up, Upstream interface protocol up
(Dwn): Downstream interface admin down, Upstream interface protocol down
Tracking (downstream)  Tracked (upstream)          Admin State   Minlink
Interfaces
-----
Te 1/0/10 (Dwn)        Te 1/0/20 (Dwn), Te 1/0/15 (Dwn)  Enabled      4
                        Te 1/0/14 (Dwn), Te 1/0/11 (Dwn)
Te 4/0/20 (Dwn)        Te 4/0/23 (Dwn), Te 4/0/22 (Dwn)  Disabled     2
                        Te 4/0/21 (Dwn)
```

## History

Release version	Command history
6.0.1	This command was introduced.
6.0.1a	The Minlink output field was added.

## Related Commands

[track \(LST\)](#)

# show tunnel

Displays statistics for tunnels in several formats; brief, detailed, and an overview.

## Syntax

```
show tunnel ID [ rbridge-id rbridge-id ]
show tunnel [ site name ] brief [ rbridge-id rbridge-id ]
show tunnel dst-ip dst_ip_address brief [ rbridge-id rbridge-id ]
show tunnel mode vxlan brief [ rbridge-id rbridge-id ]
show tunnel nsx service-node [ all [ rbridge-id rbridge-id ] | rbridge-id rbridge-id ]
show tunnel overlay-gateway name brief [ rbridge-id rbridge-id ]
show tunnel src-ip src_ip_address brief [ rbridge-id rbridge-id ]
show tunnel statistics [ rbridge-id rbridge-id ]
show tunnel brief admin-state{ up | down}[ rbridge-id rbridge-id ]
show tunnel brief oper-state { up | down} [ rbridge-id rbridge-id ]
show tunnel brief bfd-state { up | down} [ rbridge-id rbridge-id ]
```

## Parameters

### tunnel *ID*

Specifies one tunnel ID for which to show statistics. Range is 1 to 65535.

### *rbridge-id rbridge-id*

Specifies an RBridge ID.

Comma delimiters and ranging (with a hyphen) are allowed. The RBridge ID range is from 1 through 239.

### *site name*

Specifies a site that represents a remote VCS Fabric or the other end of the tunnel.

### brief

Displays brief listings for all tunnels.

### *dst-ip dst\_ip\_address*

Filters statistics by tunnel destination IP address.

### mode

Filters statistics by tunnel mode; in Network OS, the only supported mode is vxlan.

### *vxlan*

Filters statistics on all VXLANs.

### *nsx service-node*

Filters BUM-enabled tunnels to NSX service nodes.

### *all*

Displays all tunnels to NSX service nodes.



**overlay-gateway** *name*

Filter by gateway name.

**src-ip** *src\_ip\_address*

Filters statistics by tunnel source IP address.

**statistics**

Displays packet information for all tunnels.

**admin-state**

Filters packet information based on admin state.

**oper-state**

Filters packet information based on operation state.

**bfd-state**

Filters packet information based on BFD state.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command lists statistics for all the tunnels in the VCS. The output includes the tunnel ID, source IP address, destination IP address, VRF, administration state, and operational state.

For **show tunnel** *ID*, details of the specified tunnel are shown. Output includes the tunnel ID, tunnel IF index, administration state, operational state, source IP address, gateway (if any), destination IP address, packet count, byte count, and current outgoing path.

For **show tunnel statistics**, you receive packet information for all tunnels.

The following example shows brief listings for all tunnels in the VCS.

```
device# show tunnel brief
Number of tunnels: 3

Tunnel 1, mode VXLAN
Admin state: Up, Oper state: Up
Source IP 10.10.10.1, Vrf default
Destination IP 150.1.1.1

Tunnel 200, mode VXLAN
Admin state: Up, Oper state: Down
Source IP 100.1.1.11, Vrf default
Destination IP 160.1.1.1

Tunnel 300, mode VXLAN
Admin state: Up, Oper state: Up
Source IP 100.1.1.11, Vrf default
Destination IP 170.1.1.1
```

The following example displays statistics for a tunnel site.

```
device# show tunnel brief site VCS_2
Tunnel 61441, mode VXLAN, rbridge-ids 10
Admin state UP, Oper state UP
Source IP 10.2.2.1, Vrf default-vrf
Destination IP 10.1.1.1
```

The following example shows details for the tunnel with the ID of 61441.

```
device# show tunnel 61441
Tunnel 61441, mode VXLAN, rbridge-ids 1
Ifindex 2080436225, Admin state up, Oper state down
Overlay gateway "test", ID 1
Source IP 1.1.1.1 ( Loopback 1 ), Vrf default-vrf
Destination IP 10.10.10.10
Configuration source Site
MAC learning enabled
Active next hops on rbridge 1: (none)
Packet count: RX 0 TX 0
Byte count : RX (NA) TX 0
Time since last interface status change 2d18h30m
```

The following example shows high-level statistics for all tunnels.

```
device# show tunnel statistics
```

Tunnel ID	Packets TX	Packets RX	Bytes TX	Bytes RX
1	22200	2272	1982888	11000
200	2233	888922	22333	7867822

The following example shows summarized tunnel information, including BFD status.

```
device# show tunnel brief
Number of tunnels: 1

Tunnel 61441, mode VXLAN, rbridge-ids 1-2
Admin state up, Oper state up, BFD up
Source IP 50.50.50.3, Vrf default-vrf
Destination IP 20.20.20.251
```

## History

Release version	Command history
6.0.1	This command was modified to include configured BFD status.
7.0.1	The output was updated to include "number of tunnels" and "time since last interface change".

# show tunnel replicator

Displays status and details for all broadcast, unicast, and unknown multicast (BUM) forwarders present in a VCS Fabric enabled for VLAN redistribution across VXLAN NSX replicator tunnels.

## Syntax

```
show tunnel replicator [ rbridge-id rbridge-id ]
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies an RBridge ID.

Comma delimiters and ranging (with a hyphen) are allowed. The RBridge ID range is from 1 through 239.

## Modes

Privileged EXEC mode

## Examples

The following example displays status and details for all BUM forwarders present in VLAN redistribution across VXLAN NSX replicator tunnels.

```
device# show tunnel replicator

Tunnel 61442, mode VXLAN, rbridge-ids 3-4
Ifindex 2080436226, Admin state up, Oper state up, BFD up
Overlay gateway "GW2", ID 1
Source IP 20.20.20.20 ( Ve 22, Vrid 1 ), Vrf default-vrf
Destination IP 20.20.0.197
Configuration source VTEP Controller
BUM vlans: 10,31-50
  Active next hops on rbridge 3:
    IP: 20.20.20.196, Vrf: default-vrf
    Egress L3 port: Ve 22, Outer SMAC: 0027.f8da.8d77
    Outer DMAC: 0027.f83a.3499
    Egress L2 Port: Po 2, Outer ctag: 22, stag:0, Egress mode: Local
    BUM forwarder: yes

  Active next hops on rbridge 4:
    IP: 20.20.20.196, Vrf: default-vrf
    Egress L3 port: Ve 22, Outer SMAC: 0027.f8d0.abbb
    Outer DMAC: 0027.f83a.3499
    Egress L2 Port: Po 2, Outer ctag: 22, stag:0, Egress mode: Local
    BUM forwarder: no

Packet count: RX 30525 TX 0
Byte count : RX (NA) TX 0

Tunnel 61443, mode VXLAN, rbridge-ids 3-4
Ifindex 2080436227, Admin state up, Oper state up, BFD up
Overlay gateway "GW2", ID 1
Source IP 20.20.20.21 ( Ve 22, Vrid 1 ), Vrf default-vrf
Destination IP 20.20.0.198
Configuration source VTEP Controller
BUM vlans: 11-30,51,55
  Active next hops on rbridge 3:
    IP: 20.20.20.198, Vrf: default-vrf
    Egress L3 port: Ve 22, Outer SMAC: 0027.f8da.8d77
    Outer DMAC: 0027.f83a.3499
    Egress L2 Port: Po 2, Outer ctag: 22, stag:0, Egress mode: Local
    BUM forwarder: yes

  Active next hops on rbridge 4:
    IP: 20.20.20.196, Vrf: default-vrf
    Egress L3 port: Ve 22, Outer SMAC: 0027.f8d0.abbb
    Outer DMAC: 0027.f83a.3499
    Egress L2 Port: Po 2, Outer ctag: 22, stag:0, Egress mode: Local
    BUM forwarder: no
```

## History

Release version	Command history
7.0.1	This command was introduced. It replaces the <b>show tunnel nsx service-node</b> command.

# show tunnel status

Displays statistics for tunnels in tabular format.

## Syntax

```
show tunnel status ID [ rbridge-id rbridge-id ]
show tunnel status [ site name ] brief [ rbridge-id rbridge-id ]
show tunnel status src-ip src_ip_address [ rbridge-id rbridge-id ]
show tunnel status dst-ip dst_ip_address [ rbridge-id rbridge-id ]
show tunnel status mode vxlan [ rbridge-id rbridge-id ]
show tunnel status overlay-gateway name [ rbridge-id rbridge-id ]
show tunnel status admin-state{ up | down}[ rbridge-id rbridge-id ]
show tunnel status oper-state { up | down} [ rbridge-id rbridge-id ]
show tunnel status bfd-state { up | down} [ rbridge-id rbridge-id ]
```

## Parameters

### tunnel *ID*

Specifies one tunnel ID for which to show statistics. Range is 1 to 65535.

### **rbridge-id** *rbridge-id*

Specifies an RBridge ID.

Comma delimiters and ranging (with a hyphen) are allowed. The RBridge ID range is from 1 through 239.

### **site** *name*

Specifies a site that represents a remote VCS Fabric or the other end of the tunnel.

### **dst-ip** *dst\_ip\_address*

Filters statistics by tunnel destination IP address.

### **mode**

Filters statistics by tunnel mode; in Network OS, the only supported mode is vxlan.

### **vxlan**

Filters statistics on all VXLANs.

### **overlay-gateway** *name*

Filter by gateway name.

### **src-ip** *src\_ip\_address*

Filters statistics by tunnel source IP address.

### **statistics**

Displays packet information for all tunnels.

### **admin-state**

Filters packet information based on admin state.

### **oper-state**

Filters packet information based on operation state.

show tunnel status

### bfd-state

Filters packet information based on BFD state.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command lists statistics for all the tunnels in the VCS. The tabular output includes the tunnel ID, source IP address, destination IP address, VRF, administration state, and operational state.

## Examples

Typical command output.

```
device# show tunnel status
```

Tnl id	Adm state	Oper state	BFD state	Tnl dest IP
61441	up	up	up	20.20.0.127
61442	up	up	up	20.20.0.128
61443	up	up		20.20.90.2
61444	up	up		20.20.90.1
61445	up	up	down	20.20.0.197

## History

Release version	Command history
6.0.1	This command was modified to include configured BFD status.

# show udd

Shows global UDLD information.

## Syntax

```
show udd
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays global unidirectional link detection (UDLD) protocol configuration values such as whether the protocol is enabled on the switch and the *hello* time and timeout values.

## Examples

The following example displays global UDLD information for the device.

```
device# show udd
UDLD Global Information
  Admin State:      UDLD enabled
  UDLD hello time:  500 milliseconds
  UDLD timeout:    2500 milliseconds
```

## Related Commands

[protocol udd](#)

# show uddl interface

Display unidirectional link detection (UDLD) protocol information for the specified interface.

## Syntax

```
show uddl interface [ <N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

<N> **gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot/port*

Specifies a valid slot and port number.

## Modes

Privileged EXEC mode.

## Usage Guidelines

The following describes the values that appear in the output headings for this command.

**TABLE 25** Description UDLD headings

Heading	Description
State	Describes if UDLD is enable or disabled.
Mode	Describes if the mode is Receive, Transmit, or Both (Transmit/Receive).
Advertise Transmitted	Describes how often the advertisement is transmitted.
Hold time for advertise	Describes the hold time for receiving devices before discarding.
Re-init Delay Timer	The timer for the reinitializing delay
Tx Delay Timer	The timer for transmission
DCBX Version	The current DCBX version
Auto-Sense	States whether Auto-Sense is active.
Transmit TLVs	Describes what information is being transmitted for the TLV.
DCBX FCoE Priority Bits	Describes the current FCoE priority bit for DCBX.



## Examples

To display UDLD information for a specific 10-gigabit Ethernet interface:

```
device# show udd interface tengigabitethernet 5/0/1
Global Admin State: UDLD enabled
UDLD information for TenGigabitEthernet 5/0/1
  UDLD Admin State:          Enabled
  Interface Operational State: Link is down
  Remote hello time:         Unknown
  Local system id: 0x1ecd7bfa Remote system id: Unknown
  Local port : 5/0/1         Remote port : Unknown
  Local link id: 0x0         Remote link id: Unknown
  Last Xmt Seq Num: 1       Last Rcv Seq Num: Unknown
```

## Related Commands

[protocol udd](#)

# show uddl statistics

Shows UDLD statistics.

## Syntax

```
show uddl statistics [ interface { <N>gigabitethernet rbridge-id/slot/port } ]
```

## Parameters

<N> **gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command displays all unidirectional link detection (UDLD) protocol statistics or shows the statistics on a specified port

## Examples

To show UDLD statistics on a specific 10-gigabitEthernet interface:

```
device# show uddl statistics interface tengigabitethernet 5/0/1
UDLD Interface statistics for TenGigaBitEthernet 5/0/1
Frames transmitted: 310
Frames received: 301
Frames discarded: 0
Frames with error: 0
Remote port id changed: 0
Remote MAC address changed: 0
```

## Related Commands

[protocol uddl](#)

# show users

Displays the users logged in to the system and locked user accounts.

## Syntax

```
show users [ rbridge-id { rbridge-id | all } ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

### *rbridge-id*

Specifies an RBridge ID.

### all

Specifies all RBridges.

## Modes

Privileged EXEC mode

RBridge ID mode

## Examples

The following example displays active user sessions and locked user accounts.

```
device# show users
  rbridge-id
  all
**USER SESSIONS**
ID  Username   Role   Host Ip      Method   Time Logged In   TTY
2   jsmith     user   192.0.2.0    cli      2016-04-30 01:59:35 pts/2
1   jdoe       admin  192.0.2.1    cli      2016-05-30 01:57:41 tty80

**LOCKED USERS**
RBridge ID   Username
no locked users
```

## Related Commands

[show running-config username](#)

# show vcs

Displays the Extreme VCS Fabric configuration.

## Syntax

```
show vcs [ detail | virtual-ip | virtual-ipv6 ]
```

## Parameters

### detail

Displays detailed information about each RBridge in the fabric.

### virtual-ip

Displays the virtual IPv4 address.

### virtual-ipv6

Displays the virtual IPv6 address.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the Extreme VCS Fabric parameters (VCS ID and the switch RBridge ID).

## Examples

The following example displays the VCS summary view for a switch.

```
device# show vcs

Config Mode      : RBridge ID
VCS ID           : 1
Total Number of Nodes      : 1
Rbridge-Id WWN      Management IP  VCS Status Fabric Status HostName
1          10:00:00:05:33:51:63:42* 10.17.37.154 Online      Online      sw0
```

The following example displays VCS configuration details for a local-only switch.

```
device# show vcs detail

Config Mode      : Local-Only
VCS ID          : 1
Total Number of Nodes      : 6
Node :1
  Serial Number : BKN2501G00R
  Condition     : Good
  Status        : Connected to Cluster
  VCS Id        : 1
  Rbridge-Id    : 38
  Co-ordinator  : NO
  WWN           : 10:00:00:05:33:52:2A:82
  Switch MAC    : 00:05:33:52:2A:82
  FCF MAC       : 0B:20:B0:64:10:27
  Switch Type   : BR-VDX6720-24-C-24
  Internal IP   : 127.1.0.38
  Management IP : 10.17.10.38
Node :2
  Serial Number : BZA0330G00P
  Condition     : Good
  Status        : Connected to Cluster
  VCS Id        : 1
  Rbridge-Id    : 80*
  Co-ordinator  : NO
  WWN           : 10:00:00:05:33:78:00:00
  Switch MAC    : 00:05:33:78:00:81
  FCF MAC       : 19:30:00:48:19:31
  Switch Type   : VDX 8770-4
  Internal IP   : 127.1.0.80
  Management IP : 10.17.11.80
Node :3
  Serial Number : BWW2516G01G
  Condition     : Good
  Status        : Connected to Cluster
  VCS Id        : 1
  Rbridge-Id    : 82
  Co-ordinator  : NO
  WWN           : 10:00:00:05:33:6F:2B:D2
  Switch MAC    : 00:05:33:6F:2B:D2
  FCF MAC       : 0B:20:B0:64:10:26
  Switch Type   : Elara2f
  Internal IP   : 127.1.0.82
  Management IP : 10.17.10.82
(Output truncated)
```

The following example displays the VCS summary view for a switch that is in logical chassis cluster mode ("Distributed").

```
device# show vcs

device Mode      : Distributed
VCS Mode         : Logical Chassis
VCS ID          : 10
VCS GUID         : 6990c885-efa2-4b27-bb46-669d2d4d79f1
Total Number of Nodes      : 3
Rbridge-Id WWN           Management IP  VCS Status Fabric Status HostName
-----
2          10:00:00:05:33:E6:BD:00  10.24.84.203  Online      Online      sw0
3          >10:00:00:05:33:65:63:64*  10.24.83.99   Maintenance Online      sw0
4          10:00:00:27:F8:86:C2:45   10.24.83.97   Offline     Unknown     sw0
```

The following example displays VCS configuration details for distributed mode.

```
device# show vcs detail

Config Mode      : Distributed
VCS Mode         : Logical Chassis
VCS ID          : 149
VCS GUID         : 00000000000000000000000000000000
Virtual IP       : 20.0.0.1/24
Virtual IPV6     : 203::607/64
Associated rbridge-id : 2
Total Number of Nodes : 1
Nodes Disconnected from Cluster : 0
Cluster Condition : Good
Cluster Status   : All Nodes Present in the Cluster
Node :1
  Serial Number : CPL2548J00Y
  Condition     : Good
  VCS Status    : Co-ordinator
  VCS Id        : 149
  Rbridge-Id    : 2*
  Co-ordinator  : YES
  WWN           : 10:00:00:27:F8:C3:C4:B2
  Switch MAC    : 00:27:F8:C3:C4:B2
  FCF MAC       : DE:AD:BE:EF:DE:AD
  Switch Type   : BR-VDX6740
  Internal IP   : 127.1.0.2
  Management IP : 10.37.18.150
  Fabric Status : Online
  Maintenance Mode : ON
```

The following example issues the **show vcs** command on a VCS-disabled switch.

```
device# show vcs

state      : Disabled
```

The following example displays the virtual IP address.

```
device# show vcs virtual-ip

Virtual IP           : 10.21.87.2/20
Associated rbridge-id : 2
oper-status:        down
```

## History

Release version	Command history
7.0.0	The output of this command was updated to reflect maintenance mode.

# show vcs auto-config-backup

Displays the status of a running-configuration master file backup across all nodes in the cluster, whether this is configured automatically (by means of the **vcs auto-config-backup timer** command), or manually (by means of the **auto-config-backup** command).

## Syntax

```
show vcs auto-config-backup
```

## Modes

Privileged EXEC mode

## Examples

To display the status of a running-configuration file backup across all nodes in the cluster, in this example when the backup is initiated automatically (by "timer"):

```
device# show vcs auto-config-backup
Last Backup Time: 2016-08-03 20:51:11
Initiated By: timer
```

To display the status of a running-configuration file backup across all nodes in the cluster, in this example when the backup is initiated manually (by "user"):

```
device# show vcs auto-config-backup
Last Backup Time: 2016-08-03 21:32:02
Initiated By: admin (user)
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show version

Displays the current firmware version.

## Syntax

```
show version [ rbridge-id { rbridge-id | all } ] [ all-partitions ] [ brief ]
```

## Parameters

### rbridge-id

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

### all-partitions

Displays firmware information for both the active and the standby partitions. For each module, both partitions are displayed.

### brief

Displays a brief version of the firmware information.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display firmware version information and build dates. The default command output includes the following information:

- Network Operating System Version—The firmware version number
- Firmware name—The label of the firmware image
- Build Time—The build date and time of the firmware
- Install time—The date and time of the firmware installation
- Kernel—The Linux kernel version
- Boot-Prom—The size of the boot programmable read-only memory
- Control Processor—The control processor model and memory

When executed on the active management module, this command displays firmware versions on both management modules and interface modules. When executed on the standby management module, only the firmware versions for the standby management module are displayed.

The **rbridge-id** and **all** operands are not supported.



## Examples

To display the firmware version information for all partitions:

```
device# show version all-partitions

Network Operating System Software
Network Operating System Version: 7.0.0
Copyright (c) 2017-2018 Extreme Networks, Inc.
Firmware name:      3.0.0
Build Time:         01:18:17 May 26, 2018
Install Time:       10:16:24 May 27, 2018
Kernel:             2.6.34.6
BootProm:           1.0.0
Control Processor:  e500mc with 7168 MB of memory
-----
Slot   Name      Primary/Secondary Versions      Status
-----
M1     NOS       7.0.0                          STANDBY
        7.0.0
M2     NOS       7.0.0                          ACTIVE*
        7.0.0
L1/0   NOS       7.0.0                          ACTIVE
        7.0.0
L1/1   NOS       7.0.0                          STANDBY
        7.0.0
L2/0   NOS       7.0.0                          ACTIVE
        7.0.0
L2/1   NOS       7.0.0                          STANDBY
        7.0.0
```

To display the firmware for all partitions in the brief view:

```
device# show version all-partitions brief

Slot   Name      Primary/Secondary Versions      Status
-----
M1     NOS       7.0.0                          STANDBY
        7.0.0
M2     NOS       7.0.0                          ACTIVE*
        7.0.0
L1/0   NOS       7.0.0                          ACTIVE
        7.0.0
L1/1   NOS       7.0.0                          STANDBY
        7.0.0
L2/0   NOS       7.0.0                          ACTIVE
        7.0.0
L2/1   NOS       7.0.0                          STANDBY
        7.0.0
```

## Related Commands

[firmware download](#), [show firmwaredownloadstatus](#)

# show version peripheral phy

Displays the current firmware (microcode) version installed on PHY chips on an Extreme VDX 6740T or VDX 6740T-1G switch.

## Syntax

`show version peripheral phy`

## Modes

Privileged EXEC mode

## Examples

Typical command display output:

```
device# show version peripheral phy
1.38.c1
device#
```

## History

Release version	Command history
7.1.0	This command was introduced.

# show virtual-fabric status

Displays the status of the Virtual Fabric (VF): VF-capable, VF-incapable, or VF-enabled.

## Syntax

```
show virtual-fabric status
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the status of the VF with respect to all nodes in the fabric. The possible states are as follows:

- VF-capable: All nodes in the fabric can support service or transport VFs.
- VF-incapable: At least one node in the fabric cannot support service or transport VFs.
- VF-enabled: The Virtual Fabric is already enabled and service or transport VFs are supported

## Examples

```
Typical command output display.
switch# show virtual-fabric status
Fabric is virtual-fabric incapable
Rbridge-Id      Virtual-fabric status
=====
1               capable
2               capable
3               incapable
4               capable
```

# show vlag-partner-info

Displays partner information about a specific partner port-channel of a dynamic vLAG.

## Syntax

```
show vlag-partner-info [ port-channel number ]
```

## Parameters

**port-channel** *number*

Specifies a LAG port-channel number. Range is from 1 through 6144.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The output of this command displays the following information:

Output field	Description
Port-channel	The partner port-channel number.
RBridge	The RBridge ID.
Partner System ID	The partner system ID with information such as system priority and system MAC address.
Key	The partner operand key.

## Examples

The following example displays partner information for port-channel 45.

```
device# show vlag-partner-info port-channel 45
Port-channel 45
  RBridge 1: Partner System ID - 0xffff,00-05-33-48-71-a8, Key 0009
  RBridge 4: Partner System ID - 0xffff,11-22-33-44-55-66, Key 0009
  RBridge 5: Partner System ID - 0xffff,00-05-33-48-71-a8, Key 0009
```

## History

Release version	Command history
7.0.0	This command was introduced.

# show vlan

Displays information about one or more VLAN interfaces.

## Syntax

```
show vlan [ vlan_id | brief [ provisioned | unprovisioned ] | classifier ]
```

## Parameters

*vlan\_id*

Specifies the VLAN interface to display.

**brief**

Displays VLAN information for all interfaces including static and dynamic.

**classifier**

Displays all VLAN classification information.

**provisioned**

Displays provisioned VLANs.

**unprovisioned**

Displays unprovisioned VLANs.

## Modes

Privileged EXEC mode

## Examples

The following example displays information about an 802.1Q VLAN:

```
switch# show vlan 1

VLAN      Name                State      Ports
(u)-Untagged, (t)-Tagged
(c)-Converged
=====
1         default            ACTIVE    Te 0/0(t)
                           Te 0/8(t)
                           Po 1(t)
```

The following example shows all VLANs that are configured, provisioned (active) and unprovisioned (inactive):

```
switch# show vlan brief

Total Number of VLANs configured: 6
Total Number of VLANs unprovisioned: 0
Total Number of VLANs provisioned: 6
VLAN      Name          State      Ports          Classification
(F)-FCoE                                     (u)-Untagged,
(T)-Transparent                             (t)-Tagged
(R)-RSPAN                                    (c)-Converged
=====
300        vlan300        ACTIVE     Te 4/0/1(t)
5000(T)    vlan5000       ACTIVE     Te 2/0/1/(t)   ctag 50, 60, 100-200
                                           Te 4/0/1(t)     ctag 50, 60, 100-200
5500(T)    vlan5500       ACTIVE     Te 3/0/1/(t)   ctag 1, 1002, 4093, 4095
5800       vlan5800       ACTIVE     Te 2/0/1(t)    ctag 800
6000(T)    vlan6000       ACTIVE     Te 4/0/1/(t))
```

The following example shows only provisioned VLANs:

```
switch# show vlan brief provisioned

Total Number of VLANs configured: 8
Total Number of VLANs unprovisioned: 3
Total Number of VLANs provisioned: 5
VLAN      Name          State      Ports          Classification
(F)-FCoE                                     (u)-Untagged,
                                                (t)-Tagged
(R)-RSPAN                                    (c)-Converged
=====
1          default       ACTIVE     Te 2/0/5(c)
5000      VLAN5000      ACTIVE     Te 2/0/5(t)     ctag 100
                                           Te 2/0/6(u)     ctag 200
                                           Te 3/0/4(u)
6000      VLAN6000      ACTIVE     Te 3/0/5(u)     mac 0004.0004.0004
                                           Te 2/0/5(t)     ctag 300
                                           Te 3/0/5(u)     mac 0002.0002.0002
                                           Te 3/0/5 (u)    mac-group 1
7000      VLAN7000      ACTIVE     Po 10(t)        ctag 300
                                           Te 2/0/5 (t)    ctag 400
                                           Te 3/0/5 (u)    mac 0006.0006.0006
                                           Te 3/0/5 (u)    mac-group 2
1002(F)   VLAN1002      ACTIVE     Te 2/0/16(t)
                                           Te 3/0/15(t)
```

The following example shows only unprovisioned VLANs:

```
switch# show vlan brief unprovisioned

Total Number of VLANs configured: 8
Total Number of VLANs unprovisioned: 3
Total Number of VLANs provisioned: 5
VLAN      Name          State      Ports          Classification
(F)-FCoE                                     (u)-Untagged, (t)-Tagged
(R)-RSPAN                                    (c)-Converged
=====
2000      VLAN2000      INACTIVE (unprovisioned)
4000      VLAN4000      INACTIVE (unprovisioned)
8000      VLAN8000      INACTIVE (unprovisioned)
```

# show vlan brief

Displays basic information about switch VLAN interfaces.

## Syntax

```
show vlan brief [ provisioned | unprovisioned ]
```

## Parameters

### provisioned

Displays provisioned VLANs.

### unprovisioned

Displays unprovisioned VLANs.

## Modes

Privileged EXEC mode

## Usage Guidelines

You can filter to display only provisioned or unprovisioned VLANs.

## Command Output

The **show vlan brief** command displays the following information:

Output field	Description
VLAN	Displays the VLAN ID.
Name	Displays one of the following strings: <ul style="list-style-type: none"> <li>"default"</li> <li>A name assigned to the VLAN by means of the <b>name</b> command</li> <li>A default name automatically assigned to the VLAN, composed of "VLAN" and the VLAN ID. For example, if the VLAN ID is 1000, the default name is VLAN1000.</li> </ul>
State	Displays "ACTIVE" for provisioned VLANs or "INACTIVE" for unprovisioned VLANs.
Ports	Uplink ports are mapped with actual VLANs allowed on the interface. protected ports are with internal VLANs.
Classification	(Available only for provisioned VLANs).

## Examples

The following example displays output for a configuration that includes one 802.1Q VLAN and one global VLAN (GVLAN), with details as listed below.

VLAN	Description
2441	802.1Q
7561	Internal VLAN mapping for 2441
5001	GVLAN (with ctag 3001)
8073	Internal VLAN mapping for 5001 (with ctag 3001)

All interfaces displayed with internal VLANs are considered as protected ports (41/3/12 and 41/3/30). Other interfaces mapped to external VLANs 2441 for 5001 are considered as uplink ports (41/3/1, 41/3/8, 41/3/14, and 41/3/31).

```
device# show vlan brief
Total Number of VLANs configured      : 1026
Total Number of VLANs provisioned    : 1026
Total Number of VLANs unprovisioned  : 0
VLAN      Name                State                Ports                Classification
(F)-FCoE                                     (u)-Untagged
(R)-RSPAN                                     (c)-Converged
(T)-TRANSPARENT                             (t)-Tagged
=====
1          default              ACTIVE              Te 41/3/31(t)
                                         Te 41/3/1(t)
                                         Te 41/3/8(t)
                                         Te 41/3/14(t)

2441      VLAN2441              ACTIVE              Te 41/3/31(t)
                                         Te 41/3/1(t)
                                         Te 41/3/8(t)
                                         Te 41/3/14(t)

5001      VLAN5001              ACTIVE              Te 41/3/31(t)  ctag 3001
                                         Te 41/3/1(t)  ctag 3001
                                         Te 41/3/8(t)  ctag 3001
                                         Te 41/3/14(t) ctag 3001

7561      VLAN7561              ACTIVE              Te 41/3/12(t)  ctag 2441
                                         Te 41/3/30(t) ctag 2441

8073      VLAN8073              ACTIVE              Te 41/3/12(t)  ctag 3001
                                         Te 41/3/30(t) ctag 3001
```

## History

Release version	Command history
7.2.0	The command output was modified to show protected ports status.

## Related Commands

[name \(VLAN interfaces\)](#)



# show vlan classifier

Displays information about a specific VLAN classifier group.

## Syntax

```
show vlan classifier [ group number | interface group-number | interface port-channel number | rule number | interface
<N>gigabitethernet rbridge-id/slot/port ]
```

## Parameters

### group number

Specifies the VLAN classifier group number. Valid values range from 1 through 16.

### interface group number

Specifies the VLAN classifier interface group number. Valid values range from 1 through 16.

### interface port-channel number

Specifies the VLAN classifier port-channel number. Valid values range from 1 through 63.

### rule number

Specifies the VLAN classifier rule number. Valid values range from 1 through 256.

### interface <N>gigabitethernet

Specifies a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **ten****gigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

### rbridge-id

Specifies an RBridge ID.

### slot

Specifies a valid slot number.

### port

Specifies a valid port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display information about all configured VLAN classifier groups or a specific VLAN interface group.

If a group ID is not specified, all configured VLAN classifier groups are shown. If a group ID is specified, a specific configured VLAN classifier group is shown.

# show vlan private-vlan

Displays information about private VLANs.

## Syntax

`show vlan private-vlan`

## Modes

Privileged EXEC mode

## Examples

Typical command output display.  
switch# show vlan private-vlan

Primary	Secondary	Type	Ports	Classification
=====	=====	=====	=====	=====
6000		primary	Te 4/1/17(t) Te 1/2/17(t)	ctag 10 ctag 10
6000	6001	isolated	Te 4/1/17(t) Te 2/0/17(t)	ctag 11 ctag 11
6000	6002	community	Te 4/1/17(t) Te 3/1/17(t)	ctag 12 ctag 12
6000	6003	community	Te 4/1/17(t) Te 3/1/18(t)	ctag 13 ctag 13

# show vlan rspan-vlan

Displays information about remote SPAN VLANs.

## Syntax

```
show vlan rspan-vlan
```

## Modes

Privileged EXEC mode

## Examples

```
sw0(conf-if-te-1/1/34)# do show vlan rspan-vlan
Total Number of VLANs configured      : 3
Total Number of VLANs provisioned     : 2
Total Number of VLANs unprovisioned   : 1
VLAN      Name          State                Ports          Classification
=====  =====  =====
6000 (R)  VLAN6000  INACTIVE(member port down) Te 1/1/34(t)  ctag 121
6001 (R)  VLAN6001  INACTIVE(member port down) Te 1/1/34(t)  ctag 555
```

## Related Commands

[rspan-vlan](#)

# show vnetwork

Displays virtual assets from the vCenter that are discovered on an Extreme VDX device.

## Syntax

```
show vnetwork [ datacenter [ datacenter_id | vcenter vcenter_name ] | [ diff datacenter datacenter_id ] | dvpgs [ datacenter
datacenter_id | name string ] { vcenter vcenter_name } | dvs [ datacenter datacenter_id | name string ] { vcenter
vcenter_name } | hosts [ datacenter datacenter_id | name string ] { vcenter vcenter_name } | pgs [ datacenter
datacenter_id | name string ] { vcenter vcenter_name } | vcenter status | vmpolicy [ macaddr [ datacenter datacenter_id |
mac mac_address ] { vcenter string } | vms | vss [ datacenter datacenter_id | name string ] { vcenter vcenter_name } ]
```

## Parameters

### datacenter

Displays discovered data centers.

*datacenter\_id*

Datacenter ID (a string).

**vcenter** *vcenter\_name*

Specifies a vCenter.

### diff

Displays configuration differences between the current device and the specified data center. Refer to the **vnetwork reconcile vcenter** command for corrections.

#### datacenter

Specifies a data center.

*datacenter\_id*

Datacenter ID (a string).

### dvpgs

Displays distributed virtual port groups.

**datacenter** *datacenter\_id*

Specifies a datacenter. This is optional and need not be used unless required.

**name** *string*

Specifies a distributed virtual port group.

### dvs

Displays distributed virtual devices.

**name** *string*

Selects a distributed virtual device name. This is optional and need not be used unless required.

### hosts

Displays discovered hosts.

**name** *string*

Specifies a host name.

**vcenter** *vcenter\_name*  
Specifies a vCenter (required).

**pgs**  
Displays discovered standard port groups.

**name** *string*  
Specifies a standard port group.

**vcenter status**  
Displays configured vCenter status.

**vmpolicy**  
Displays association between virtual network interface cards (vNICs) or VM kernel NICs (vmkNICs) and port groups or port profiles.

**macaddr**  
Displays policies by MAC address.

**mac** *mac\_address*  
Selects a six-octet MAC address; for example, 00:50:56:8e:00:4b.

**vms**  
Displays discovered VMs.

**vss**  
Displays discovered standard virtual devices.

**name**  
Selects a virtual device.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the virtual assets configured on the vCenter and discovered on the VDX device.

The device interface column information is local to each device in the fabric.

## Examples

```
device# show vnetwork ?
```

Possible completions:

```
datacenter Shows discovered datacenters
diff Shows vcenter and switch configuration diff
dvpgs Shows discovered distributed virtual port-groups
dvs Shows discovered distributed virtual switches
hosts Shows discovered hosts
pgs Shows discovered standard port-groups
vcenter Shows configured vCenter
vmpolicy Shows vnics/vmknics to portgroup to port-profile association
vms Shows discovered VMs
vss Shows discovered standard virtual switches
```

```
device# show vnetwork dvpgs
```

dvPortGroup	dvSwitch	Vlan
ProductionVMs	dvSwitch-Production	10-10,
dvSwitch-Production-DVUplinks-7589	dvSwitch-Production	0-4094,

```
switch# show vnetwork dvs
```

dvSwitch	Host	Uplink Name	Switch Interface
dvSwitch-Production40	-	-	-
dvSwitch-Production41	-	-	-

Total Number of Entries: 2

```
device# show vnetwork hosts
```

Host	Uplink Name	Uplink MAC	(d)Virtual Switch	Switch Interface
ESX-4921.englab.extremenetworks.com	vmnic0	e4:1f:13:43:54:90	vSwitch0	-
	vmnic2	00:1b:21:8f:4a:f0	dvSwitch-Production	115/0/5
	vmnic4	00:05:33:26:3e:ba	vSwitch1	115/0/1
	vmnic5	00:05:33:26:3e:bb	dvSwitch-Production	-
ESX-4922.englab.extremenetworks.com	vmnic0	e4:1f:13:43:95:5c	vSwitch0	-
	vmnic2	00:05:33:26:2d:90	dvSwitch-Production	115/0/10
	vmnic3	00:05:33:26:2d:91	dvSwitch-Production	115/0/11
	vmnic5	00:05:1e:eb:f9:94	vSwitch1	115/0/2

```
device# show vnetwork pgs
```

PortGroup	vSwitch	vlanId	Host
TestVMs	vSwitch1	50-50,	ESX-4922.englab.extremenetworks.com
	vSwitch1	50-50,	ESX-4921.englab.extremenetworks.com
VMkernel	vSwitch1	0-0,	ESX-4922.englab.extremenetworks.com
	vSwitch1	0-0,	ESX-4921.englab.extremenetworks.com

```
switch# show vnetwork vcenter status
```

vCenter	Start	Elapsed (sec)	Status
MYVC	2011-09-07 14:08:42	10	In progress

```
device# show vnetwork vmpolicy macaddr all
```

Associated MAC	Virtual Machine	(dv)PortGroup	Port-Profile
00:50:56:72:42:4c	-	ProductionVMs	auto-ProductionVMs
00:50:56:78:69:36	-	VMkernel	auto-VMkernel
00:50:56:7b:e5:41	-	ProductionVMs	auto-ProductionVMs
00:50:56:7d:96:16	-	VMkernel	auto-VMkernel
00:50:56:8e:00:4b	CentOS-4921	ProductionVMs	auto-ProductionVMs
00:50:56:8e:00:4d	CentOS-4921	TestVMs	auto-TestVMs
00:50:56:8e:00:50	CentOS-4922	TestVMs	auto-TestVMs
00:50:56:8e:00:51	CentOS-4922	ProductionVMs	auto-ProductionVMs

```
switch# show vnetwork vms
```

```

Virtual Machine      Associated MAC      IP Addr      Host
=====
CentOS-4921         00:50:56:8e:00:4b -          ESX-4921.englab.extremenetworks.com
                   00:50:56:8e:00:4d -          ESX-4921.englab.extremenetworks.com
CentOS-4922         00:50:56:8e:00:50 -          ESX-4922.englab.extremenetworks.com
                   00:50:56:8e:00:51 -          ESX-4922.englab.extremenetworks.com
vSwitch             Host                Uplink Name   Switch Interface
=====
vSwitch0            djesxi-5064.englab.extremenetworks.com vmnic0        -
                   ht-153.englab.extremenetworks.com    vmnic1        -
                   ht-153.englab.extremenetworks.com    vmnic0        -
vSwitch1            ht-153.englab.extremenetworks.com    vmnic7        -
                   ht-154.englab.extremenetworks.com    vmnic6        -
                   ht-154.englab.extremenetworks.com    vmnic7        -
vSwitch2            ht-153.englab.extremenetworks.com    vmnic6        -
Total Number of Entries: 8

```

## History

Release version	Command history
5.0.2b	This command was introduced.
6.0.0	This command description was clarified.

# show vrf

Displays Virtual Routing and Forwarding (VRF) configuration information.

## Syntax

```
show vrf [ vrf-name | detail | interface ] [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*vrf-name*

Specifies a named VRF. For the default VRF, enter **default-vrf**.

**detail**

Displays detailed information for all VRFs configured.

**interface**

Displays VRF information for an interface that you specify.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Examples

The following example displays basic information for the default VRF.

```
device# show vrf default-vrf
VRF-Name: default-vrf, VRF-Id: 1
IP Router-Id: 50.50.50.1
Interfaces:
    Ve 40, Ve 84, Ve 85, Ve 150, Ve 211,
    Ve 501, Ve 503, Ve 504, Ve 505, Ve 1025,
    Ve 1059, Ve 2000, Lo 50
Address-family IPv4 unicast
    Max routes: -   Route count:134
    No import route-maps
    No export route-maps
Address-family IPv6 unicast
    Max routes: -   Route count:51
    No import route-maps
    No Export route-maps
```



The following example displays basic information for all VRFs.

```
device# show vrf
Total number of VRFs configured: 4
VrfName      VrfId  V4-Ucast  V6-Ucast
blue         3      Enabled   -
default-vrf  1      Enabled   Enabled
mgmt-vrf     0      Enabled   Enabled
red          2      -         Enabled
```

The following example displays detailed information for all VRFs.

```
device# show vrf detail
Total number of VRFs configured: 4

VRF-Name: blue, VRF-Id: 3
IP Router-Id: 10.1.1.10
Interfaces:
  Ve 200
Address-family IPV4 unicast
  Max routes:-   Route count:134
  No import route-maps
  No export route-maps

VRF-Name: default-vrf, VRF-Id: 1
IP Router-Id: 30.1.1.1
Interfaces:
  Ve 300
Address-family IPV4 unicast
  Max routes:-   Route count:51
  No import route-maps
  No export route-maps

Address-family IPV6 unicast
  Max routes:-   Route count:2
  No import route-maps
  No Export route-maps

VRF-Name: mgmt-vrf, VRF-Id: 0
IP Router-Id: 0.0.0.0
Interfaces:
  mgmt 1, Null0
Address-family IPV4 unicast
  Max routes:-   Route count:3
  No import route-maps
  No export route-maps

Address-family IPV6 unicast
  Max routes:-   Route count:2
  No import route-maps
  No Export route-maps

VRF-Name: red, VRF-Id: 2
IP Router-Id: 0.0.0.0
Interfaces:
  Ve 100
Address-family IPV6 unicast
  Max routes:-   Route count:2
  No import route-maps
  No Export route-maps
```

The following example displays basic VRF information for a specified RBridge.

```
device# show vrf rbridge-id 11
Total number of VRFs configured: 4
VrfName      VrfId  V4-Ucast  V6-Ucast
blue         3      Enabled   -
default-vrf  1      Enabled   Enabled
mgmt-vrf     0      Enabled   Enabled
red          2      -         Enabled
```

show vrf

The following example indicates which VRFs are available on which interfaces.

```
device# show vrf interface
VrfName      Interfaces
blue         Ve 200
default-vrf  Ve 300
mgmt-vrf     mgmt 1, Null0
red          Ve 100
```

## Related Commands

[address-family unicast \(VRF\)](#), [max-route](#), [vrf](#)

# show vrrp

Displays information about IPv4 VRRP and VRRP-E sessions.

## Syntax

```
show vrrp
show vrrp VRID [ detail | summary ] [ rbridge-id { rbridge-id | all } ]
show vrrp detail [ rbridge-id { rbridge-id | all } ]
show vrrp summary [ vrf { vrf-name | all } | rbridge-id { rbridge-id | all } ]
show vrrp summary vrf default-vrf
show vrrp interface { <N>gigabitethernet [ rbridge-id / slot / port | port-channel number ] [ detail | summary ]
show vrrp interface ve vlan_id [ detail | summary ] [ rbridge-id { rbridge-id | all } ]
show vrrp rbridge-id { rbridge-id | all }
```

## Parameters

### *VRID*

The virtual group ID about which to display information. The range is from 1 through 255.

### detail

Displays all session information in detail, including session statistics.

### summary

Displays session-information summaries.

### vrf

Specifies a VRF instance or all VRFs.

#### *vrf-name*

Specifies a VRF instance. For the default vrf, enter **default-vrf**.

### all

Specifies all VRFs.

### rbridge-id

Specifies an RBridge or all RBridges.

#### *rbridge-id*

Specifies an RBridge.

### all

Specifies all RBridges.

### interface

Displays information for an interface that you specify.

### <N> gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

*rbridge-id*  
(Optional) Specifies an RBridge ID.

*slot*  
Specifies a valid slot number.

*port*  
Specifies a valid port number.

**port-channel** *number*  
Specifies a port-channel interface. The range is from 1 through 6144.

**ve** *vlan\_id*  
Specifies the VE VLAN number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display information about VRRP and VRRP-E sessions, either in summary or full-detail format. You can also specify a particular virtual group or interface for which to display output.

This command is for VRRP and VRRP-E. VRRP-E supports only the VE interface type. You can modify or redirect the displayed information by using the default Linux tokens ([, >).

To display information for VRRP sessions using the default VRF, you can use the **show vrrp summary** command syntax (with no additional parameters).

For the default or a named VRF, you can use the **show vrrp summary vrf** command syntax with the *vrf-name* option.

To display information for all VRFs, use the **show vrrp summary vrf all** command.

## Examples

The following example shows all VRRP/VRRP-E session information in detail, including session statistics.

```
device# show vrrp detail
=====Rbridge-id:4=====

Total number of VRRP session(s)   : 2

VRID 18
Interface: Ve 2018; Ifindex: 1207961570
Mode: VRRP
Admin Status: Enabled
Description :
Address family: IPv4
Version: 2
Authentication type: No Authentication
State: Master
Session Master IP Address: Local
Virtual IP(s): 18.1.1.100
Virtual MAC Address: 0000.5e00.0112
Configured Priority: unset (default: 100); Current Priority: 100
Advertisement interval: 1 sec (default: 1 sec)
Preempt mode: ENABLE (default: ENABLE)
Hold time: 0 sec (default: 0 sec)
Master Down interval: 4 sec
Trackport:
  Port(s)                Priority  Port Status
  =====                =====  =====

Global Statistics:
=====
Checksum Error : 0
Version Error  : 0
VRID Invalid   : 0

Session Statistics:
=====
Advertisements           : Rx: 0, Tx: 49
Gratuitous ARP           : Tx: 1
Session becoming master  : 1
Advts with wrong interval : 0
Prio Zero pkts           : Rx: 0, Tx: 0
Invalid Pkts Rvcd        : 0
Bad Virtual-IP Pkts      : 0
Invalid Authentication type : 0
Invalid TTL Value        : 0
Invalid Packet Length    : 0

VRID 19
Interface: Ve 2019; Ifindex: 1207961571
Mode: VRRP
Admin Status: Enabled
Description :
Address family: IPv4
Version: 2
Authentication type: No Authentication
State: Master
Session Master IP Address: Local
Virtual IP(s): 19.1.1.100
Virtual MAC Address: 0000.5e00.0113
Configured Priority: unset (default: 100); Current Priority: 100
Advertisement interval: 1 sec (default: 1 sec)
Preempt mode: ENABLE (default: ENABLE)
Hold time: 0 sec (default: 0 sec)
Master Down interval: 4 sec
Trackport:
  Port(s)                Priority  Port Status
  =====                =====  =====

Global Statistics:
```

show vrrp

```
=====  
Checksum Error : 0  
Version Error : 0  
VRID Invalid   : 0  
  
Session Statistics:  
=====  
Advertisements      : Rx: 0, Tx: 81  
Gratuitous ARP      : Tx: 1  
Session becoming master : 1  
Advts with wrong interval : 0  
Prio Zero pkts      : Rx: 0, Tx: 0  
Invalid Pkts Rvcd   : 0  
Bad Virtual-IP Pkts : 0  
Invalid Authenticon type : 0  
Invalid TTL Value   : 0  
Invalid Packet Length : 0
```

The following example displays summary information for VRRP/VRRP-E statistics on the VRF named blue.

```
device# show vrrp summary vrf blue  
=====Rbridge-id:4=====
```

Total number of VRRP session(s)	: 1
Master session count	: 1
Backup session count	: 0
Init session count	: 0

VRID	Session	Interface	Admin State	Current Priority	State	Short-path Forwarding	Revert Priority	SPF Reverted
====	=====	=====	=====	=====	=====	=====	=====	=====
18	VRRP	Ve 2018	Enabled	100	Master			

The following example displays summary information for VRRP/VRRP-E statistics on all VRFs.

```
device# show vrrp summary vrf all  
  
=====Rbridge-id:4=====
```

Total number of VRRP session(s)	: 2
Master session count	: 2
Backup session count	: 0
Init session count	: 0

VRID	Session	Interface	Admin State	Current Priority	State	Short-path Forwarding	Revert Priority	SPF Reverted
====	=====	=====	=====	=====	=====	=====	=====	=====
18	VRRP	Ve 2018	Enabled	100	Master			
19	VRRP	Ve 2019	Enabled	100	Master			

The following example displays summary information for VRRP/VRRP-E statistics on the default VRF. (This command is equivalent to **show vrrp summary**.)

```
device# show vrrp summary vrf default-vrf  
=====Rbridge-id:4=====
```

Total number of VRRP session(s)	: 1
Master session count	: 1
Backup session count	: 0
Init session count	: 0

VRID	Session	Interface	Admin State	Current Priority	State	Short-path Forwarding	Revert Priority	SPF Reverted
====	=====	=====	=====	=====	=====	=====	=====	=====
19	VRRP	Ve 2019	Enabled	100	Master			

The following example displays information for VRRP-E tracked networks.

```

device# show vrrp detail

=====
Rbridge-id:1
=====

Total number of VRRP session(s)   : 1

VRID 3
  Interface: Ve 100;  Ifindex: 1207959652
  Mode: VRRPE
  Admin Status: Enabled
  Description :
  Address family: IPv4
  Version: 2
  Authentication type: No Authentication
  State: Master
  Session Master IP Address: Local
  Virtual IP(s): 10.1.1.100
  Virtual MAC Address: 02e0.523d.750a
  Configured Priority: unset (default: 100); Current Priority: 100
  Advertisement interval: 1 sec (default: 1 sec)
  Preempt mode: DISABLE (default: DISABLED)
  Advertise-backup: DISABLE (default: DISABLED)
  Backup Advertisement interval: 60 sec (default: 60 sec)
  Short-path-forwarding: Disabled
  Revert-Priority: unset; SPF Reverted: No
  Hold time: 0 sec (default: 0 sec)
  Master Down interval: 4 sec
  Trackport:
    Port(s)                Priority  Port Status
    =====                =====  =====
  Tracknetwork:
    Network(s)              Priority  Status
    =====                =====  =====
    10.20.1.0/24            50       Up

Global Statistics:
=====
  Checksum Error : 0
  Version Error  : 0
  VRID Invalid   : 0

Session Statistics:
=====
  Advertisements           : Rx: 0, Tx: 35
  Neighbor Advertisements  : Tx: 19
  Session becoming master  : 1
  Advts with wrong interval : 0
  Prio Zero pkts           : Rx: 0, Tx: 0
  Invalid Pkts Rcvd        : 0
  Bad Virtual-IP Pkts      : 0
  Invalid Authentication type : 0
  Invalid TTL Value        : 0
  Invalid Packet Length    : 0
  VRRPE backup advt sent   : 0
  VRRPE backup advt recvd  : 0

```

## History

Release version	Command history
6.0.1	This command was modified to add output to verify the VRRP-E track network feature.
7.0.0	This command was modified to support port-channels.

show vrrp

## Related Commands

[clear vrrp statistics](#), [show ipv6 vrrp](#)



# show zoning enabled-configuration

Displays information about the enabled zoning configuration.

## Syntax

```
show zoning enabled-configuration
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the `cfg-name` and `enabled-zone` fields of the enabled zoning configuration for an Extreme VCS Fabric.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric.

## Examples

```
device# show zoning enabled-configuration
zoning enabled-configuration cfg-name cfg1
zoning enabled-configuration enabled-zone zone1
  member-entry 10:00:00:00:00:00:00:01
zoning enabled-configuration enabled-zone zone2
  member-entry 10:00:00:00:00:00:00:02

device# show zoning enabled-configuration | count
Count: 8 lines
```

## Related Commands

[show running-config zoning defined-configuration](#), [show running-config zoning enabled-configuration](#)

# show zoning operation-info

Displays information about transactions and database size.

## Syntax

```
show zoning operation-info
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric, but it is always executed on the principal R Bridge.

## Command Output

The **show zoning operation-info** command displays the following information:

Output field	Description
db-max	Defines the maximum size in bytes of the zone database in nonvolatile memory.
db-avail	Displays the size in bytes of the unused portion of nonvolatile memory available for the defined configuration.
db-committed	Displays the size in bytes of the defined configuration currently stored in nonvolatile memory.
db-transaction	Displays the size in bytes of the uncommitted defined configuration.
db-token	Displays the current transaction token ID.
last-zone-changed-timestamp	Displays the last time the defined zone configuration was modified.
last-zone-committed-timestamp	Displays the data and time of the last time the zoning database was saved to nonvolatile memory.

## Examples

To display information about transactions and database size:

```
device# show zoning operation-info

db-max 1045274
db-avail 1043822
db-committed 440
db-transaction 0
transaction-token 0
last-zone-changed-timestamp 2011-11-16 14:38:15 GMT-7:00
last-zone-committed-timestamp 2011-11-16 14:38:15 GMT-7:00
```

## Related Commands

show running-config zoning defined-configuration, show running-config zoning enabled-configuration, show zoning enabled-configuration



# Commands shutdown through Z

---

## shutdown

Disables the selected interface.

## Syntax

**shutdown**

**no shutdown**

## Command Default

The interface is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no shutdown** to enable the interface.

When an RBridge is rejoining the logical chassis cluster, the interface-level configuration is reset to the default values.

If you use in-band management only, you may choose to shut down the management interface (which is considered *out of band*). When the management interface is shut down, all services (such as ping, scp, telnet, ssh, snmp, firmwaredownload, and supportsave) through the management interface IP on the active MM are unavailable. The chassis VIP and VCS VIP align with the active MM on a chassis system.

Management interface shutdown is a persistent configuration, meaning that the interface remains down after a system reboot or failover.

## Examples

To disable a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 1/0/1
switch(conf-if-te-1/0/1)# shutdown
```

To enable a specific 1-gigabit Ethernet interface:

```
switch(config)# interface gigabitethernet 1/0/2
switch(conf-if-gi-1/0/2)# no shutdown
```

shutdown

To disable a specific management interface:

```
switch(config)# interface Management 1/0
```

```
switch(conf-Management-1/0)# shutdown
```

# shutdown (STP)

Disables Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), Multiple Spanning Tree Protocol (MSTP), Per-VLAN Spanning Tree+ (PVST+), or Rapid PVST+ (R-PVST+) globally.

## Syntax

**shutdown**

**no shutdown**

## Command Default

STP is not enabled as it is not required in a loop-free topology.

## Modes

Any of the supported spanning tree configuration modes (STP, RSTP, MSTP, PVST+, R-PVST+)

## Usage Guidelines

Enter **no shutdown** to re-enable any of the supported versions of STP.

## Examples

To disable RSTP globally:

```
device# configure terminal
device(config)# protocol spanning-tree rstp
device(conf-rstp)# shutdown
```

To enable MSTP globally:

```
device# configure terminal
device(config)# protocol spanning-tree mstp
device(conf-mstp)# no shutdown
```

# shutdown (UDLD)

Disables the unidirectional link detection (UDLD) protocol on all ports without affecting configuration.

## Syntax

**shutdown**

**no shutdown**

## Modes

Protocol UDLD configuration mode

## Usage Guidelines

The **no shutdown** command unblocks all ports that have been blocked by the UDLD protocol.

## Examples

To shutdown the UDLD protocol:

```
switch# configure
switch(config)# protocol udld
switch(config-udld)# shutdown
```

## Related Commands

[protocol udld](#)



# shutdown (VXLAN)

Administratively shuts down tunnels to a VXLAN overlay gateway site.

## Syntax

**shutdown**

**no shutdown**

## Modes

VXLAN overlay gateway site configuration mode

## Usage Guidelines

The **no shutdown** command enables tunnels to the site.

The "no shutdown" state for this mode is not displayed in the running configuration.

## Examples

To shut down VXLAN overlay gateway tunnels:

```
switch(config)# overlay-gateway gateway1
switch(config-overlay-gw-gateway1)# site mysite
switch(config-overlay-gw-gateway1-site-mysite)# shutdown
```

## Related Commands

[overlay-gateway](#), [site](#)

# shutdown-time

Specifies the delay between the time a port is disabled after Edge Loop Detection (ELD) detects a loop and the automatic re-enabling of that port.

## Syntax

```
shutdown-time num
```

```
no shutdown-time
```

## Command Default

The default value is 0 minutes.

The port is not re-enabled automatically.

## Parameters

*num*

Specifies the number of minutes before a port is re-enabled. Valid values range from 0 through 1440 minutes (1 minutes to 24 hours. Specify 0 to disable the automatic reenabling of that port).

## Modes

ELD configuration mode

## Usage Guidelines

### NOTE

Any change to **shutdown-time** only takes effect for the ports that are disabled by ELD after the configuration change. Any ports that were already disabled by ELD before the **shutdown-time** change continues to follow the old **shutdown-time** value. These ports start to follow the new shutdown time after the currently running timer expires and ELD still detects the loop and shuts down the port again.

If you do not set a shutdown time using this command, you can re-enable all ELD-disabled ports manually using the **clear edge-loop-detection** command.

Enter **no shutdown-time** to return to the default value.

## Examples

To re-enable ports 24 hours after they are disabled by ELD:

```
switch(config)# protocol edge-loop-detection
switch(config-eld)# shutdown-time 1440
```

To cancel automatic port re-enable:

```
switch(config-eld)# no shutdown-time
```

## Related Commands

[clear edge-loop-detection](#), [show edge-loop-detection globals](#), [show edge-loop-detection rbridge-id](#)

## site

Creates a remote Layer 2 extension site in a VXLAN overlay gateway context and enables VXLAN overlay gateway site configuration mode.

## Syntax

**site** *name*

**no site** *name*

## Parameters

*name*

Site identifier. An ASCII character string up to 63 characters long, including the alphabet, numbers 0 through 9, hyphens (-), and underscores (\_).

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

The VXLAN overlay gateway type must first be configured for Layer 2 extension, by means of the **type layer2-extension** command.

A "site" represents a remote VCS Fabric or the other end of the VXLAN tunnel. A site is associated with a "container," as data structure that includes the destination IPv4 address of the tunnel, the switchport VLANs, and the administrative state.

Use the **no site** command with a specified name to remove the tunnel that corresponds to the site. Once you create the site instance, you enter VXLAN overlay gateway site configuration mode, where you can configure other properties for the site. The key commands available in this mode are summarized below:

**TABLE 26** Key commands available in VXLAN overlay gateway site configuration mode

Command	Description
<b>bfd</b>	Configures Bidirectional Forwarding Detection (BFD) on a tunnel in VXLAN overlay gateway configurations.
<b>bfd interval</b>	Configures BFD session parameters on a tunnel in VXLAN overlay gateway configurations.
<b>extend vlan</b>	Configures switchport VLANs for the tunnels to the containing site in a VXLAN overlay gateway configurations.
<b>ip address</b>	Specifies the IPv4 address of a destination tunnel in VXLAN overlay gateway configurations.
<b>mac-learning protocol bgp</b>	Changes the default MAC learning on a tunnel from Layer 2 to BGP MAC learning.
<b>shutdown</b>	Administratively shuts down tunnels to a VXLAN overlay gateway site.

## Examples

To create a VXLAN overlay gateway site and enter VXLAN overlay gateway site configuration mode:

```
switch(config)# overlay-gateway gateway1
switch(config-overlay-gw-gateway1)# site mysite
switch(config-overlay-gw-gateway1-site-mysite)#
```

## History

Release version	Command history
6.0.1	Support for the <b>bfd</b> and <b>bfd interval</b> commands was added in VXLAN overlay gateway site configuration mode.

# slot

Enables or disables the slot.

## Syntax

`slot number { disable | enable }`

## Command Default

This command has no defaults.

## Parameters

*number*

The slot to control. The valid values are 0 or 1.

**disable**

Disables the slot.

**enable**

Enables the slot.

## Modes

Privileged EXEC mode

# snmp-server community

Sets the community string and associates it with the user-defined group name to restrict the access of MIB for SNMPv1 and SNMPv2c requests.

## Syntax

```
snmp-server community string [ groupname group-name ] [ ipv4-acl standard-ipv4-acl-name ] [ ipv6-acl standard-ipv6-acl-name ]
```

```
no snmp-server community string [ groupname group-name ] [ ipv4-acl standard-ipv4-acl-name ] [ ipv6-acl standard-ipv6-acl-name ]
```

## Command Default

By default no group name is mapped with the community string. User must map the community string with any non-existing or existing group name to contact the switch through SNMPv1 or SNMPv2c.

## Parameters

*string*

Specifies the community name string. The number of characters available for the string ranges from 1 through 64.

**groupname** *group-name*

Specifies the group name associated with the community name.

**ipv4-acl** *standard-ipv4-acl-name*

Specifies an IPv4 ACL that contains rules permitting or denying access from specified IPv4 addresses.

**ipv6-acl** *standard-ipv6-acl-name*

Specifies an IPv6 ACL that contains rules permitting or denying access from specified IPv6 addresses.

## Modes

Global configuration mode

## Usage Guidelines

For the Extreme VDX 2746, only the default group names (admin and user) are supported.

This command manages the configuration of the SNMP agent in the device. The configuration includes SNMPv1 and SNMPv2c configuration settings.

The maximum number of SNMP communities supported is 256.

Use a **no** form of this command to do one of more of the following:

- Remove the specified community string and all entities associated with it
- Remove the groupname from the string
- Remove the IPv4 ACL from the string
- Remove the IPv6 ACL from the string

## Examples

The following example adds the community string "public" and associates the group name "user" with it.

```
device(config)# snmp-server community public groupname user
device(config)#
```

The following example also applies an IPv4 ACL and an IPv6 ACL.

```
device(config)# snmp-server community comm1 groupname accGroup1 ipv4-acl standV4ACL1 ipv6-acl
standV6ACL1
device(config)#
```

The following example removes the IPv4 and IPv6 ACLs from the "public" community.

```
device(config)# no snmp-server community public ipv4-acl
device(config)# no snmp-server community public ipv6-acl
```

## History

Release version	Command history
5.0.2	This command was modified to apply ACLs that contains rules permitting or denying access from specified addresses.
6.0.2	This command was modified to extend the community name string to 64 characters.



# snmp-server contact

Sets the SNMP server contact string.

## Syntax

`snmp-server contact string`

`no snmp-server contact string`

## Command Default

The default contact string is "Field Support."

## Parameters

*string*

Specifies the server contact. You must enclose the text in double quotes if the text contains spaces.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default values.

## Examples

To set the SNMP server contact string to "Operator 12345":

```
switch(config)# snmp-server contact "Operator 12345"
```

To set the SNMP server contact string to the default of "Field Support":

```
switch(config)# no snmp-server contact
```

# snmp-server context

Maps the context name in an SNMPv3 packet's protocol data unit (PDU) to the name of a VPN routing and forwarding (VRF) instance.

## Syntax

```
snmp-server context context_name vrf-name vrf_name
no snmp-server context context_name vrf-name vrf_name
```

## Parameters

*context-name*

Enables the specification of a variable *context\_name* that can be passed in the SNMP PDU.

**vrf-name** *vrf\_name*

Specifies a variable that can be retrieved when an SNMP request is sent with the configured *context\_name*. This variable can be used in SNMP requests for "ipCidrRouteTable."

## Modes

Global configuration mode

## Usage Guidelines

The context-to-VRF mapping is one-to-one and is applicable to all SNMP versions. Only one context is allowed per VRF instance.

### ATTENTION

SNMP SET requests work only on the default VRF.

## Examples

To map the context name "mycontext" to the VRF name "myvrf":

```
device(config)# snmp-server context mycontext vrf-name myvrf
```

To delete the VRF name "myvrf":

```
device(config)# no snmp-server context mycontext vrf-name myvrf
```

To create the new VRF name "mynewvrf" and map the context to it:

```
device(config)# snmp-server context mycontext vrf-name mynewvrf
```

## History

Release version	Command history
7.1.0	This command was modified to update the Usage Guidelines.

# snmp-server enable trap

Enables the SNMP traps.

## Syntax

`snmp-server enable trap`

`no snmp-server enable trap`

## Command Default

The SNMP server traps are enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no** form of this command to disable the SNMP traps.

## Examples

The following example disables the SNMP traps.

```
device# configure terminal
device(config)# no snmp-server enable trap
```

The following example enables the SNMP traps.

```
device# configure terminal
device(config)# snmp-server enable trap
```

## History

Release version	Command history
5.0.0	This command was introduced.

# snmp-server engineid local

Configures an SNMP engine ID for the SNMP agent.

## Syntax

```
snmp-server engineid local engine_id  
no snmp-server engineid local
```

## Command Default

A default engine ID is generated during system start up.

## Modes

RBridge ID configuration mode

## Usage Guidelines

A reboot is necessary for the configured engine ID to become active.

Use the **no** form of the command to remove the configured engine ID from database.

## Examples

The following example configures an engine ID for the SNMP agent.

```
device(config-rbridge-id-152)# snmp-server engineid local 10:00:00:05:33:51:A8:65:05:33:51:A8
```

The following example removes the configured engine ID from the database.

```
device (config-rbridge-id-152)# no snmp-server engineid local
```

## snmp-server group

Creates user-defined groups for SNMPv1/v2/v3 and configures read, write, and notify permissions to access the MIB view.

### Syntax

```
snmp-server group groupname { v1 | v2c | v3 { auth | noauth | priv } } [ read viewname ] [ write viewname ] [ notify viewname ]
```

```
no snmp-server group groupname { v1 | v2c | v3 { auth | noauth | priv } } [ read viewname ] [ write viewname ] [ notify viewname ]
```

### Command Default

None

### Parameters

*groupname*

Specifies the name of the SNMP group to be created.

**v1 | v2c | v3**

Specifies the version of SNMP.

**auth | noauth | priv**

Specifies the various security levels for SNMPv3.

**auth**

Specifies the authNoPriv security level. Password authentication is used based on either MD5 or SHA hash authentication and no encryption is used for communications between the devices.

**noauth**

Specifies the noAuthNoPriv security level. If no security level is specified, noauth is the default. This security level means that there is no authentication password exchanged and the communications between the agent and the server are not encrypted. The SNMP requests are authorized based on a username string match similar to the community string for SNMPv1/v2c.

**priv**

Specifies the authPriv security level. Password authentication is used based on either MD5 or SHA hash authentication and the communication between the agent and the server are also encrypted.

**read** *viewname*

Specifies the name of the view that enables you to provide read access.

**write** *viewname*

Specifies the name of the view that enables you to provide both read and write access.

**notify** *viewname*

Specifies the name of the view that enables you to provide access to the MIB for trap or inform.

## Modes

Global configuration mode

## Usage Guidelines

Maximum number of SNMP groups supported is 10.

This command is not supported on the Extreme VDX 2746.

## Examples

The following example creates SNMP server group entries for SNMPv3 user group with auth or noauth permission.

```
device(config)# snmp-server group group1 v3 auth read myview write myview notify myview
device(config)# snmp-server group group2 v3 noauth read all write all notify all
device(config)# snmp-server group group3 v3 auth
```

The following example removes the configured SNMP server groups.

```
device(config)# no snmp-server group test1 v3 auth
device(config)# no snmp-server group TEST1 v3 auth read myview write myview
device(config)# no snmp-server group TEST2 v3 noauth read all write all notify all
```

# snmp-server host

Configures the SNMP trap server host attributes.

## Syntax

```
snmp-server host { ipv4_host | ipv6_host | dns_host } community_string [ version { 1 | 2c } ] [ udp-port port ] [ severity-level |
{ none | debug | info | warning | error | critical } ] [ source-interface { loopback number | ve vlan_id } ] [ use-vrf vrf-name ]
```

```
no snmp-server host { ipv4_host | ipv6_host | dns_host } community_string [ version { 1 | 2c } ] [ udp-port port ] [ severity-
level | { none | debug | info | warning | error | critical } ] [ source-interface { loopback number | ve vlan_id } ] [ use-vrf vrf-
name]
```

## Parameters

**host** { ipv4\_host | ipv6\_host | dns\_host }

Specifies the IP address of the host. IPv4, IPv6, and DNS hosts are supported.

*community\_string*

Specifies the community string associated with the host entry. The number of characters available for the string ranges from 1 through 64.

**version** { 1 | 2c }

Selects version 1 or 2c traps to be sent to the specified trap host.

**udp-port** *port*

Specifies the UDP port where SNMP traps will be received. Valid port IDs range from 0 through 65535. The default port is 162.

**severity-level** { none | debug | info | warning | error | critical }

Provides the ability to filter traps based on severity level on both the host and the SNMPv3 host. Only RASLog (swEvent) traps can be filtered based on severity level. The configured severity level marks the reporting threshold. All messages with the configured severity or higher are displayed. If the severity level of **none** is specified, all traps are filtered and no RASLog traps are received.

**source-interface**

Replaces the default SNMP source IP address with any loopback or VE interface IP address as a source IP address for SNMP notification.

**loopback** *number*

Specifies to display the loopback interface number. Valid values range from 1 through 255.

**ve** *vlan\_id*

Specifies a virtual Ethernet (VE) interface (VLAN interface number).

**use-vrf** *vrf-name*

Specifies a VRF through which to communicate with the SNMP host. By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Modes

Global configuration mode

## Usage Guidelines

This command sets the trap destination IP addresses and SNMP version, associates a community string with a trap host community string (for v1 and v2c), and specifies the UDP destination port where SNMP traps will be received.

To configure SNMP trap hosts associated with community strings, you must create the community string using the **snmp-server community** command before configuring the host.

The host supports six communities and their associated trap recipients and trap recipient severity levels. The default value for the trap recipient of each community is 0.0.0.0. The length of the community string should be between 2 and 64 characters.

The **no snmp-server host host community-string string version 2c** command brings version 2c down to version 1.

The **no snmp-server host host community-string string** command removes the SNMP server host from the device configuration altogether.

## Examples

The following example creates an entry for trap host 1050:0:0:0:5:600:300c:326b associated with community "public." The trap host receives traps from the configured device.

```
device(config)# snmp-server host 1050:0:0:0:5:600:300c:326b public severity-level Info
```

The following example creates an entry for trap host brcd.extremenetworks.com associated with community "public." The trap host receives traps from the configured device.

```
device(config)# snmp-server host brcd.extremenetworks.com public severity-level info
```

The following example associates "commaccess" as a read-only community and set 10.32.147.6 as a trap recipient with SNMP version 2c on target port 162.

```
device(config)# snmp-server host 10.32.147.6 commaccess version 2c udp-port 162
```

The following example creates a trap host (10.23.23.45) associated with the community "public", which will receive all traps with the severity level of Info.

```
device(config)# snmp-server host 10.23.23.45 public severity-level info
```

The following example resets the severity level to None.

```
device(config)# snmp-server host 10.23.23.45 public severity-level none
```

The following example specifies a VRF to communicate with the host.

```
device(config)# snmp-server host 10.24.61.10 public use-vrf myvrf
```

## History

Release version	Command history
6.0.1	This command was modified to support the <b>use-vrf</b> keyword.
6.0.2	This command was modified to support the <b>source-interface</b> keyword.
7.0.0	This command was modified to accept any user-specified VRF.



# snmp-server location

Sets the SNMP server location string.

## Syntax

`snmp-server location string`

`no snmp-server location string`

## Command Default

The location string is "End User Premise."

## Parameters

*string*

Specifies the SNMP server location string. You must enclose the text in double quotes if the text contains spaces.

## Modes

Global configuration mode

## Examples

To set the SNMP server location string to "Building 3 Room 214":

```
switch(config)# snmp-server location "Building 3 Room 214"
```

To set the SNMP server location to the default, "End User Premise":

```
switch(config)# no snmp-server location
```

# snmp-server mib community-map

Maps an SNMP community string to an SNMP context.

## Syntax

```
snmp-server mib community-map community-name context context-name
no snmp-server mib community-map community-name context context-name
```

## Command Default

None

## Parameters

*community-name*  
Specifies an SNMP community name.

**context** *context-name*  
Specifies an SNMP context.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no** form of this command to remove a community string and its associated context name.

Any incoming SNMPv1/v2c requests with the specified community name uses the context name specified by this command. The context name can be used in SNMP requests for "ipCidrRouteTable." One community can be mapped to only one context. However, a single context can be mapped to multiple communities.

Before mapping the community to context, a valid context should be configured by using the **snmp-server context** command and a valid community string should be configured by using the **snmp-server community** command.

## Examples

The following example maps an SNMP community string to a context name.

```
device# configure terminal
device(config)# snmp-server mib community-map public context mycontext
```

The following example removes an SNMP community string and its associated context name.

```
device(config)# no snmp-server mib community-map public context mycontext
```

## History

Release version	Command history
7.0.0	This command was introduced.

# snmp-server sys-descr

Sets the Management Information Base (MIB-2) object identifier (OID) system description.

## Syntax

```
snmp-server sys-descr string  
no snmp-server sys-descr
```

## Command Default

The system description is "Extreme VDX switch".

## Parameters

*string*

The text for the system description. The string must be between 4 and 255 characters in length.

## Modes

Global configuration mode

## Usage Guidelines

Enter **no snmp-server sys-descr** to return to the default system description.

## Examples

To set the system description OID to "Extreme Cluster switch":

```
switch(config)# snmp-server sys-descr "Extreme Cluster switch"
```

To restore the system description OID to the default:

```
switch(config)# no snmp-server sys-descr
```

# snmp-server three-tuple-if enable

Configures whether the ifDescr and ifName objects that belong to the Interfaces Group MIB (IF-MIB) are represented in 2-tuple or 3-tuple format.

## Syntax

```
snmp-server three-tuple-if enable
no snmp-server three-tuple-if enable
```

## Command Default

This option is disabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

If the option is enabled, then the ifDescr and ifName objects are represented as 3-tuple. If it is disabled (the default), then these objects are represented as 2-tuple.

If this option is enabled, use the **no** form of this command to disable it and represent options as 2-tuple.

The results of this command will appear in the running configuration only when it is enabled.

Following an upgrade to Network OS 7.0.1, this option is disabled by default, and the ifDescr and ifName objects are represented as 2-tuple.

## Examples

To enable the representation of ifDescr and ifName objects as 3-tuple:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# snmp-server three-tuple-if enable
```

To disable the representation of ifDescr and ifName objects as 3-tuple and return to the default (2-tuple):

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# no snmp-server three-tuple-if enable
```

## History

Release version	Command history
7.0.1	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

## snmp-server user

Creates or changes the attributes of SNMPv3 users, and allows the SNMPv3 user to be associated with the user-defined group name.

### Syntax

```
snmp-server user username [ groupname group-name ] [ auth { md5 | sha | noauth } ] [ auth-password string [ encrypted ] ]
  [ priv { DES | AES128 | nopriv } ] [ priv-password string [ encrypted ] ] [ ipv4-acl standard-ipv4-acl-name ] [ ipv6-acl
  standard-ipv6-acl-name ]

no snmp-server user username [ groupname group-name ] [ auth { md5 | sha | noauth } ] [ auth-password string
  [ encrypted ] ] [ priv { DES | AES128 | nopriv } ] [ priv-password string [ encrypted ] ] [ ipv4-acl standard-ipv4-acl-name ]
  [ ipv6-acl standard-ipv6-acl-name ]
```

### Command Default

None

### Parameters

#### *username*

The name of the user that connects to the agent. The name must be between 1 and 16 characters long.

#### **groupname** *group-name*

The name of the group to which the user is associated. The configured user is allowed to be associated with the user-defined groups created using the **snmp-server group** command.

#### **auth**

Initiates an authentication level setting session. The default level is **noauth**.

#### **noauth**

Specifies "No Authentication Protocol".

#### **md5**

The HMAC-MD5-96 authentication level.

#### **sha**

The HMAC-SHA-96 authentication level.

#### **auth-password** *string*

A string that enables the agent to receive packets from the host. Passwords are plain text and must be added each time for each configuration replay. The password must be between 1 and 32 characters long.

#### **priv**

Initiates a privacy authentication level setting session. The default level is **nopriv**.

#### **DES**

Specifies the DES privacy protocol.

#### **AES128**

Specifies the AES128 privacy protocol.

**nopriv**

Specifies "No Privacy Protocol".

**priv-password** *string*

Specifies a string (not to exceed 32 characters) that enables the host to encrypt the contents of the message that it sends to the agent. Passwords are plain text and must be added each time for each configuration replay. The privacy password alone cannot be configured. You configure the privacy password with the authentication password.

**encrypted**

Encrypts the input for auth/priv passwords. The encrypted key should be used only while entering the encrypted auth/priv passwords.

**ipv4-acl** *standard-ipv4-acl-name*

Specifies an IPv4 ACL that contains rules permitting or denying access from specified IPv4 addresses.

**ipv6-acl** *standard-ipv6-acl-name*

Specifies an IPv6 ACL that contains rules permitting or denying access from specified IPv6 addresses.

## Modes

Global configuration mode

RBridge ID configuration mode

## Usage Guidelines

For the VDX 2746, only the default group names (snmpadmin and snmpuser) are supported.

This command configures SNMPv3 users that can be associated with a trap and inform response functionality. This command also allows configured user to be associated with user-defined SNMP groups created using the **snmp-server group** command. The maximum number of SNMP users that can be configured is 10. Optional encryption for **auth-password** and **priv-password** is also provided.

When creating a new SNMPv3 user without group name, by default there is no group name mapped with the SNMPv3 user. You must map the configured SNMPv3 user with any non-existing or existing group name available in the group CLI configuration to contact the device through SNMPv3.

The behavior of this command in the local RBridge ID configuration is same as the global configuration. If the user name configured is same in both global and RBridge ID configurations, then the RBridge ID configuration will take precedence. The encrypted password generated in the global configuration can be used for another global user to modify the passwords. The encrypted passwords generated in global configurations cannot be used in the RBridge ID configurations and vice versa.

This command may not be successful where encrypted passwords are generated by third-party or open-source tools.

Use a **no** form of this command to do one of more of the following:

- Remove the specified user and all entities associated with it
- Remove the groupname from the user
- Remove the IPv4 ACL from the user
- Remove the IPv6 ACL from the user

## Examples

The following example configures a basic authentication policy.

```
device(config)# snmp-server user extreme groupname snmpadmin auth md5 auth-password user123 priv AES128
priv-password user456
```

The following example configures plain-text passwords.

```
device(config)# snmp-server user snmpadmin1 auth md5 auth-password private123 priv DES priv-password
public123
```

The following example configures encrypted passwords.

```
device(config)# snmp-server user snmpadmin2 groupname snmpadmin auth md5 auth-password "MVB
+360X3kcfBzug5Vo6dQ==\n" priv DES priv-password "ckJFoHbzVvhR0xFRPjsMTA==\n" encrypted
```

The following example creates the SNMP users "user1" and "user2" associated with user-defined group "group1" under global configuration mode.

```
device(config)# snmp-server user user1 groupname group1
device(config)# snmp-server user user2 groupname group1 auth md5 auth-password password priv DES priv-
password password
```

The following example configures an SNMPv3 user under local RBridge ID configuration mode.

```
device(config-rbridge-id-1)# snmp-server user snmpadmin1 groupname snmpadmin auth sha auth-password
private123 priv DES priv-password public123
```

The following example configures the SNMPv3 users "user1" and "user2" associated with user-defined group "group1" under global configuration mode. It also applies an IPv4 ACL and an IPv6 ACL to "user1."

```
device(config)# snmp-server user user1 groupname group1 ipv4-acl standV4ACL1 ipv6-acl standV6ACL1
device(config)# snmp-server user user2 groupname group1 auth md5 auth-password password priv DES priv-
password
```

The following example removes groupname, the authentication and privacy protocols, and the IPv4 ACL from the user.

```
device(config)# no snmp-server user user1 groupname snmpadmin auth sha priv DES ipv4-acl
```

## History

Release version	Command history
5.0.2	This command was modified to apply ACLs that contains rules permitting or denying access from specified addresses.



# snmp-server v3host

Specifies the host recipient for SNMPv3 trap notification.

## Syntax

```
snmp-server v3host [ host { ipv4_host | ipv6_host | dns_host } ] user_name [ notifytype { traps | informs } ] engineid engine-id udp-port port_number [ severity-level { none | debug | info | warning | error | critical } ] [ source-interface { loopback number | ve vlan_id } ] [ use-vrf { vrf-name } ]
```

```
no snmp-server v3host [ host { ipv4_host | ipv6_host | dns_host } ] [ source-interface { loopback number | ve vlan_id } ] [ use-vrf { vrf-name } ]
```

## Parameters

**ipv4\_host | ipv6\_host | dns\_host**

Specifies the IP address of the host. IPv4, IPv6, and DNS hosts are supported.

**user\_name**

Specifies the SNMPv3 user name to be associated with the SNMPv3 host entry.

**notifytype traps | informs**

Specifies the type of notification traps that are sent for the host. Traps and informs are supported. The default notify type is traps.

**engineID engine-id**

Configures the remote engine ID to receive informs on a remote host.

**udp-port port\_number**

Specifies the UDP port of the host. The default UDP port number is 162.

**severity-level { none | debug | info | warning | error | critical }**

Provides the ability to filter traps based on severity level on both the host and the SNMPv3 host. Only RASLog (swEvent) traps can be filtered based on severity level. The configured severity level marks the reporting threshold. All messages with the configured severity or higher are displayed. If the severity level of None is specified, all traps are filtered and no RASLog traps are received. The default severity level is none.

**source-interface**

Replaces the default SNMP source IP address with any loopback or VE interface IP address as a source IP address for SNMP notification.

**loopback number**

Specifies to display the loopback interface number. Valid values range from 1 through 255.

**ve vlan\_id**

Specifies a virtual Ethernet (VE) interface (VLAN interface number).

**use-vrf vrf-name**

Configures SNMP to use the specified VRF to communicate with the host. The default is mgmt-vrf.

## Modes

Global configuration mode

RBridge ID configuration mode

## Usage Guidelines

You can associate a global SNMPv3 host only with global SNMPv3 users and the local SNMPv3 host only with local SNMPv3 users. You cannot create a SNMPv3 host by associating with the local SNMPv3 users and vice versa.

## Examples

The following example creates an entry for SNMPv3 trap IPv4 host 10.23.23.45 associated with SNMP user "snmpadmin1."

```
device(config)# snmp-server v3host 10.23.23.45 snmpadmin1 severity-level info
```

The following example creates an entry for SNMPv3 trap IPv6 host 1050:0:0:0:5:600:300c:326b associated with SNMP user "snmpadmin2." The trap host receives SNMPv3 traps from the configured device.

```
device(config)# snmp-server v3host 1050::5:600:300c:326b snmpadmin2 severity-level Info
```

The following example creates an entry for SNMPv3 trap host 10.26.3.166 associated with SNMP user "snmpuser2" under RBridge ID configuration mode. The trap host will receive SNMPv3 traps from the configured switch.

```
device(config-rbridge-id-1)# snmp-server v3host 10.26.3.166 snmpuser2 severity-level Info udp-port 4425
```

The following example removes the SNMPv3 trap host 10.26.3.166 associated with SNMP user "snmpuser2".

```
device(config-rbridge-id-1)# no snmp-server v3host 10.26.3.166 snmpuser2
```

The following example associates the default-vrf VRF for a trap host recipient.

```
device(config)# snmp-server v3host 10.24.61.10 public use-vrf default-vrf
```

## History

Release version	Command history
6.0.1	This command was modified to support the <b>use-vrf</b> keyword.
6.0.2	This command was modified to support the <b>source-interface</b> keyword.
7.0.0	This command was modified to support the <i>vrf-name</i> variable.

# snmp-server view

Creates a view entry with MIB object IDs to be included or excluded for user access.

## Syntax

**snmp-server view** *view-name* *mib\_tree* **included** | **excluded**

**no snmp-server view** *view-name* *mib\_tree* **included** | **excluded**

## Command Default

None

## Parameters

*view-name*

Specifies the alphanumeric name to identify the view. The name should not contain spaces.

*mib\_tree*

Specifies the MIB object ID called Object Identifiers (OIDs) that represent the position of the object or sub-tree in the MIB hierarchy.

**included** | **excluded**

Specifies whether the specified MIB object ID must be included in the view or excluded from the view.

## Modes

Global configuration mode

## Usage Guidelines

The maximum number of views supported with MIB tree entries is 10. Either a single view name associated with 10 different MIB object IDs or 10 different view names associated with each one of the MIB object IDs is allowed.

This command is not supported on the Extreme VDX 2746.

## Examples

The following example creates an SNMP view entry "view1" with excluded permission for the MIB object ID "1.3.6.1.2.1.1.3."

```
device(config)# snmp-server view view1 1.3.6.1.2.1.1.3 excluded
```

The following example creates an SNMP view entry "view2" with included permission for the MIB object ID "1.3.6.1."

```
device(config)# snmp-server view view2 1.3.6.1 included
```

The following example removes the SNMP view entry "view1" from the configuration list.

```
device(config)# no snmp-server view view1 1.3.6.1.2.1.1.3 excluded
```

# snmp-server offline-if enable

Enables the command behavior to display the interfaces belonging to an offline slot.

## Syntax

`snmp-server offline-if enable`

`no snmp-server offline-if enable`

## Command Default

Disabled by default.

## Modes

Global configuration mode.

## Usage Guidelines

Use the `no snmp-server offline-if enable` command to enable the command behavior to display the interfaces belonging to an offline slot.

## History

Release version	Command history
5.0.2	This command was introduced.

# snmp-server trap link-status

Manages the linkUp or linkDown traps under an interface sub-mode (Physical interface/Virtual Ethernet (VE) interface/ Port-Channel interface, Loopback interface).

## Syntax

**snmp-server trap-link status**

**no snmp-server trap-link status**

## Command Default

The SNMP server traps under all interfaces are enabled by default.

## Modes

Global configuration mode.

## Usage Guidelines

Use the **no snmp-server trap-link status** to disable the linkUp or linkDown traps under an interface sub-mode.

## History

Release version	Command history
5.0.2	This command was introduced.

## source

Configures the monitoring session.

## Syntax

```
source [ fortygigabitethernet rbridge-id/slot/port | <N>gigabitethernet rbridge-id/slot/port | destination | direction [ rx | tx | both ]
```

```
no source [ <N>gigabitethernet rbridge-id/slot/port | destination | direction [ rx | tx | both ]
```

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### *rbridge-id*

Specifies an RBridge ID.

#### *slot*

Specifies a valid slot number.

#### *port*

Specifies a valid port number.

### destination

Use this parameter to specify the interface.

### direction rx

Specifies to monitor the receiving traffic.

### direction tx

Specifies to monitor the transmitting traffic

### direction both

Specifies to monitor transmitting and receiving traffic.

## Modes

Monitor session configuration mode

## Usage Guidelines

Enter **no source** followed by the identifying parameters to delete the port mirroring connection for the specified interface.

## Examples

To enable session 22 for monitoring traffic:

```
switch(config)# monitor session 22
switch(config-session-22)# source tengigabitethernet 0/1 destination tengigabitethernet 0/15 direction
both
```

## Related Commands

[monitor session](#)

# span session

Configures the SPAN session.

## Syntax

**span session** *session\_id*

**no span session** *session\_id*

## Parameters

*session\_id*

Designates the session number for the flow-based SPAN session.

## Modes

Policy class configuration mode

## Usage Guidelines

Use the **no span session** *session-id* command to delete the session.

## Related Commands

[destination](#)



# spanning-tree autoedge

Enables automatic edge detection.

## Syntax

**spanning-tree autoedge**

**no spanning-tree autoedge**

## Command Default

Auto detection is not enabled.

## Modes

Interface configuration mode

## Usage Guidelines

The port can become an edge port if no Bridge Protocol Data Unit (BPDU) is received.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes. Enter **no spanning-tree autoedge** to disable automatic edge detection.

## Examples

To enable automatic edge detection:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/1)# spanning-tree autoedge
```

## Related Commands

[protocol spanning-tree](#)

# spanning-tree bpdu-mac

Sets the MAC address of the Bridge Protocol Data Unit (BPDU).

## Syntax

```
spanning-tree bpdu-mac [ 0100.0ccc.cccd | 0304.0800.0700 ]  
no spanning-tree bpdu-mac [ 0100.0ccc.cccd | 0304.0800.0700 ]
```

## Parameters

0100.0ccc.cccd  
Cisco Control Mac

0304.0800.0700  
Extreme Control Mac

## Modes

Interface configuration mode

## Usage Guidelines

This command will only take effect when the protocol is PVST+ or R-PVST+.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes.

Extreme devices support PVST+ and R-PVST+ only. The PVST and R-PVST protocols are proprietary to Cisco and are not supported.

Enter **no spanning-tree bpdu-mac 0100.0ccc.cccd** to remove the address.

## Examples

To set the MAC address of the BPDU:

```
device# configure terminal  
device(config)# interface tengigabitethernet 0/1  
device(conf-if-te-0/1)# spanning-tree bpdu-mac 0100.0ccc.cccd
```

# spanning-tree cost

Changes an interface's spanning-tree port path cost.

## Syntax

```
spanning-tree cost cost
```

```
no spanning-tree cost cost
```

## Command Default

The default path cost is 200000000.

## Parameters

*cost*

Specifies the path cost for the Spanning Tree Protocol (STP) calculations. Valid values range from 1 through 200000000.

## Modes

Interface configuration mode

## Usage Guidelines

Lower path cost indicates a greater chance of becoming root.

If xSTP is enabled over VCS, this command must be executed on all the RBridge nodes.

## Examples

To set the port cost to 128:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/1)# spanning-tree cost 128
```

## Related Commands

[show spanning-tree](#)

# spanning-tree edgeport

Enables the edge port on an interface to allow the interface to quickly transition to the forwarding state.

## Syntax

```
spanning-tree edgeport [ bpdu-filter | bpdu-guard ]  
no spanning-tree edgeport [ bpdu-filter | bpdu-guard ]
```

## Command Default

Edge port is disabled.

## Parameters

### bpdu-filter

Sets the edge port Bridge Protocol Data Unit (BPDU) filter for the port.

### bpdu-guard

Guards the port against the reception of BPDUs.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command is only for RSTP and MSTP. Use the **spanning-tree portfast** command for STP.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

Note the following details about edge ports and their behavior:

- A port can become an edge port if no BPDU is received.
- A port must become an edge port before it receives a BPDU.
- When an edge port receives a BPDU, it becomes a normal spanning-tree port and is no longer an edge port.
- Because ports directly connected to end stations cannot create bridging loops in the network, edge ports directly transition to the forwarding state, and skip the listening and learning states

## Examples

To enable a port to quickly transition to the forwarding state:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(config-if-te-0/1)# spanning-tree edgeport
```

To set the edge port BPDU filter for the port:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(config-if-te-0/1)# spanning-tree edgeport
device(config-if-te-0/1)# spanning-tree edgeport bpdu-filter
```

To guard the port against reception of BPDUs:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(config-if-te-0/1)# spanning-tree edgeport
device(config-if-te-0/1)# spanning-tree edgeport bpdu-guard
```

## Related Commands

[spanning-tree portfast](#), [spanning-tree autoedge](#)

# spanning-tree guard root

Enables the guard root to restrict which interface is allowed to be the spanning tree root port or the device's path-to-the-root.

## Syntax

```
spanning-tree guard root [ vlan vlan_id ]  
no spanning-tree guard root
```

## Command Default

Guard root is disabled.

## Parameters

**vlan *vlan\_id***  
Specifies a VLAN.

## Modes

Interface configuration mode

## Usage Guidelines

Guard root protects the root bridge from malicious attacks and unintentional misconfigurations where a bridge device that is not intended to be the root bridge becomes the root bridge. This causes severe bottlenecks in the data path. Guard root ensures that the port on which it is enabled is a designated port. If the guard root enabled port receives a superior Bridge Protocol Data Unit (BPDU), it goes to a discarding state.

If the VLAN parameter is not provided, the guard root functionality is applied globally for all per-VLAN instances. But for the VLANs which have been configured explicitly, the per-VLAN configuration takes precedence over the global configuration.

The root port provides the best path from the switch to the root switch.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes. Enter **no spanning-tree guard root** to disable guard root on the selected interface.

On the Extreme VDX family of switches, VLANs are treated as interfaces from a configuration point of view. By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). Valid VLAN IDs are as follows:

- On Extreme VDX 8770 switches: 1 through 4086 for 802.1Q VLANs (VLAN IDs 4087 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.
- On all other Extreme VDX switches: 1 through 3962 for 802.1Q VLANs (VLAN IDs 3963 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.

## Examples

To enable guard root:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/1)# spanning-tree guard root
```

## Related Commands

[show spanning-tree](#)

# spanning-tree hello-time

Configures the hello-time in seconds on the interface.

## Syntax

```
spanning-tree hello-time seconds  
no spanning-tree hello-time
```

## Command Default

2 seconds.

## Parameters

*seconds*

Sets the interval between the hello Bridge Protocol Data Units (BPDUs) sent by the root switch configuration messages. Valid values range from 1 through 10.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command sets the interval time between the BPDUs sent by the root switch. This command is only for MSTP.

Changing the **hello-time** value affects all spanning-tree instances.

The **max-age** command setting must be greater than the **spanning-tree hello-time** command setting.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes. Enter **no spanning-tree hello-time** to return to the default setting.

## Examples

To set the hello time to 5 seconds:

```
switch(config)# interface tengigabitethernet 0/1  
switch(conf-if-te-0/1)# spanning-tree hello-time 5
```

## Related Commands

[forward-delay](#), [max-age](#), [show spanning-tree](#)



# spanning-tree ieee-bpdu limit-vlan-flood

Restricts IEEE BPDU to within VLAN 4095.

## Syntax

**spanning-tree ieee-bpdu limit-vlan-flood**

**no spanning-tree ieee-bpdu limit-vlan-flood**

## Command Default

The IEEE BPDU is not restricted.

## Modes

Rbridge ID configuration mode

## Usage Guidelines

Use the **no spanning-tree ieee-bpdu limit-vlan-flood** command to remove the BPDU VLAN restriction.

When the device receives the IEEE BPDU on ingress port , it uses a ctrl-classifier entry that assigns the BPDU to the control VLAN 4095. It then broadcasts the BPDU on all edge ports, regardless of VLAN. However in an VF environment flooding should be limited. This command restricts the BPDU to VLAN 4095 and prevents flooding.

## Examples

Typical command example

```
device(config-rbridge-id-158)# spanning-tree ieee-bpdu limit-vlan-flood
```

## History

Release version	Command history
5.0.2b	This command was introduced.

# spanning-tree instance

Sets restrictions for the port of particular MSTP instances.

## Syntax

```
spanning-tree instance instance_id [ cost cost | priority priority | restricted-role | restricted-tcn ]
no spanning-tree instance instance_id
```

## Command Default

The path-cost value is 2000 on a 10-gigabit Ethernet interface.

## Parameters

*instance\_id*

Specifies the MSTP instance. Valid values range from 1 through 32.

**cost** *cost*

Specifies the path-cost for a port. Valid values range from 1 through 20000000.

**priority** *priority*

Specifies the port priority for a bridge in increments of 16. Valid values range from 0 through 240.

**restricted-role**

Specifies to restrict the role of a port.

**restricted-tcn**

Specifies to restrict the propagation of the topology change notifications from a port.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command for MSTP-specific configurations.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

Enter **no spanning-tree instance** *instance\_id* to remove the specified MSTP instance.

## Examples

To set restrictions for the port of MSTP instance 1 with the cost of 40000:

```
switch(config)# interface tengigabitethernet 0/1
switch(conf-if-te-0/1)# spanning-tree instance 1 cost 40000
```

## Related Commands

instance, show spanning-tree

# spanning-tree link-type

Enables and disables the rapid transition for the Spanning Tree Protocol (STP).

## Syntax

```
spanning-tree link-type [ point-to-point | shared ]
```

## Command Default

The **spanning-tree link-type** is set to **point-to-point**.

## Parameters

### point-to-point

Enables rapid transition.

### shared

Disables rapid transition.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command overrides the default setting of the link type.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

## Examples

To specify the link type as shared:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/0)# spanning-tree link-type shared
```

## spanning-tree peer-switch

When the Peer-Switch feature is enabled on a Cisco vPC domain, it broadcasts the same BPDUs from both vPC primary and secondary nodes to peer devices. But a VCS on a VLAG assumes that any logical interface receives only one BPDU from any of its member ports, so when it receives the two BPDUs from a Cisco vPC domain it creates a churn of VLAG mastership, and this increases the CPU load on an Extreme VDX. To avoid these problem, BPDUs received on the VLAG non-master are dropped. When the Peer-Switch functionality is enabled and the the VLAG Master is selected, BPDUs received on VLAG Non-Master are dropped unless there is a change in the status of the VLAG Master. By default, the Peer-Switch feature functionality is not active.

### Syntax

**spanning-tree peer-switch**

**no spanning-tree peer-switch**

### Command Default

By default, the Peer-Switch option is disabled.

### Modes

Interface configuration mode

### Usage Guidelines

Use the **no spanning-tree peer-switch** command to disable the Peer-Switch functionality on a port-channel. This command is only applicable on port-channel interfaces.

### Examples

Typical command example:

```
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/0)# spanning-tree peer-switch
```

To de-activate this functionality:

```
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/0)# no spanning-tree peer-switch
```

### History

Release version	Command history
6.0.0	This command was introduced.

# spanning-tree portfast

Enables the Port Fast feature on an interface to allow the interface to quickly transition to forwarding state.

## Syntax

```
spanning-tree portfast [ bpdu-filter | bpdu-guard ]
```

```
no spanning-tree portfast [ bpdu-filter | bpdu-guard ]
```

## Command Default

Port Fast is disabled.

## Parameters

### bpdu-filter

Sets the Port Fast BPDU filter for the port.

### bpdu-guard

Guards the port against the reception of BPDUs.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command is applicable the only for the Spanning Tree Protocol (STP). Port Fast immediately puts the interface into the forwarding state without having to wait for the standard forward time. Use the **spanning-tree edgeport** command for MSTP and RSTP.

BPDU filter prevents the switch from sending BPDU frames on ports that are enabled with portfast.

BPDU guard disables all portfast-enabled ports should they ever receive BPDU frames. It does not prevent transmitting of BPDU frames.

If you enable **spanning-tree portfast bpdu-guard** on an interface and the interface receives a BPDU, the software disables the interface and puts the interface in the ERR\_DISABLE state.

Enable Port Fast on ports connected to host. Enabling Port Fast on interfaces connected to switches, bridges, hubs, and so on can cause temporary bridging loops, in both trunking and nontrunking mode.

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

## Examples

To enable a port to quickly transition to the forwarding state:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(config-if-te-0/1)# spanning-tree portfast
```

To set the edge port BPDU filter for the port:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(config-if-te-0/1)# spanning-tree portfast
device(config-if-te-0/1)# spanning-tree portfast bpdu-filter
```

To guard the port against reception of BPDUs:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(config-if-te-0/1)# spanning-tree portfast
device(config-if-te-0/1)# spanning-tree portfast bpdu-guard
```

## Related Commands

[show spanning-tree](#), [spanning-tree autoedge](#), [spanning-tree edgeport](#)

# spanning-tree priority

Changes an interface's spanning-tree port priority.

## Syntax

`spanning-tree priority priority`  
`no spanning-tree priority`

## Command Default

The default value is 128.

## Parameters

*priority*

Specifies the interface priority for the spanning tree. The range of valid values is from 0 through 240. Port priority is in increments of 16.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no spanning-tree priority** to return to the default setting.

If xSTP is enabled over VCS, this command must be executed on all the RBridges.

## Examples

To configure the port priority to 16:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/1)# spanning-tree priority 16
```

## Related Commands

[show spanning-tree](#), [spanning-tree cost](#)



# spanning-tree restricted-role

Restricts the role of the port from becoming a root port.

## Syntax

```
spanning-tree restricted-role  
no spanning-tree restricted-role
```

## Command Default

The restricted role is disabled.

## Modes

Interface configuration mode

## Usage Guidelines

If xSTP is enabled over VCS, this command must be executed on all the RBridges.

Enter **no spanning-tree restricted-role** to return to the default setting.

## Examples

To configure the port from becoming a root port:

```
device# configure terminal  
device(config)# interface tengigabitethernet 0/1  
device(conf-if-te-0/1)# spanning-tree restricted-role
```

## Related Commands

[show spanning-tree](#)

# spanning-tree restricted-tcn

Restricts the Topology Change Notification (TCN) Bridge Protocol Data Units (BPDUs) sent on the port.

## Syntax

`spanning-tree restricted-tcn`

`no spanning-tree restricted-tcn`

## Command Default

The restricted TCN is disabled.

## Modes

Interface configuration mode

## Usage Guidelines

Enter `no spanning-tree restricted-tcn` to disable this parameter.

If xSTP is enabled over VCS, this command must be executed on all the RBridges.

## Examples

To restrict the TCN on a specific interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/1)# spanning-tree restricted-tcn
```

## Related Commands

[show spanning-tree](#)

# spanning-tree shutdown

Enables or disables spanning tree on the interface or VLAN.

## Syntax

**spanning-tree shutdown**

**no spanning-tree shutdown**

## Command Default

Spanning tree is disabled by default.

## Modes

Interface (Ethernet or VLAN) configuration mode

## Usage Guidelines

Enter **no spanning-tree shutdown** to enable spanning tree on the interface or VLAN.

Once all of the interfaces have been configured for a VLAN, you can enable Spanning Tree Protocol (STP) for all members of the VLAN with a single command. Whichever protocol is currently selected is used by the VLAN. Only one type of STP can be active at a time.

A physical interface (port) can be a member of multiple VLANs. For example, a physical port can be a member of VLAN 1002 and VLAN 55 simultaneously. In addition, VLAN 1002 can have STP enabled and VLAN 55 can have STP disabled simultaneously.

Vlan 1002 can not be enabled with the **spanning-tree shutdown** command.

If xSTP is enabled over VCS, this command must be executed on all the RBridges.

## Examples

To disable spanning tree on a specific interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 0/1
device(conf-if-te-0/1)# spanning-tree shutdown
```

## Related Commands

[protocol spanning-tree](#)

# spanning-tree vlan

Configures the VLAN identifier for the spanning tree interface.

## Syntax

```
spanning-tree vlan vlan_id  
no spanning-tree vlan
```

## Parameters

```
vlan vlan_id  
Sets the VLAN identifier for the spanning tree interface.
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

If xSTP is enabled over VCS, this command must be executed on all RBridge nodes.

Enter **no spanning-tree vlan** to remove the VLAN setting.

On the Extreme VDX family of switches, VLANs are treated as interfaces from a configuration point of view. By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). Valid VLAN IDs are as follows:

- On Extreme VDX 8770 switches: 1 through 4086 for 802.1Q VLANs (VLAN IDs 4087 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.
- On all other Extreme VDX switches: 1 through 3962 for 802.1Q VLANs (VLAN IDs 3963 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.

## Related Commands

[forward-delay](#), [max-age](#), [show spanning-tree](#)

# speed (Ethernet)

Sets the speed negotiation value on an Ethernet interface.

## Syntax

```
speed { 100 | 1000 | 1000-auto | 10000 | auto }  
no speed
```

## Command Default

The speed is set to **auto**.

## Parameters

**100**  
Forces the speed to 100 Mbps.

**1000**  
Forces the speed to 1 Gbps.

**1000-auto**  
Forces the speed to 1 Gbps AN (802.3 Clause 37 Auto-Negotiation)

**10000**  
Forces the speed to 10 Gbps.

**auto**  
Allows the interface to negotiate the speed setting.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of the command to reset to the default setting.

The speed command is not available for 40-gigabit Ethernet ports.

The VDX 6740 and VDX 6940 only support auto negotiate at 1 Gbps.

## Examples

To set the speed to 10 Gbps on a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 170/0/1  
switch(conf-if-te-170/0/1)# speed 10000
```

speed (Ethernet)

## Related Commands

[interface](#)

# speed (Fibre Channel)

Sets the operational speed on a Fibre Channel port.

## Syntax

```
speed { auto | 2gbps | 4gbps | 8gbps | 16gbps }
```

## Command Default

Speed is **auto** .

## Parameters

- auto**  
Allows the interface to negotiate the port speed.
- 2gbps**  
Sets the operational port speed to 2 Gbps.
- 4gbps**  
Sets the operational port speed to 4 Gbps.
- 8gbps**  
Sets the operational port speed to 8 Gbps.
- 16gbps**  
Sets the operational port speed to 16 Gbps.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The Fibre Channel version of this command can be used only on Network OS platforms with Fibre Channel Flexports.

## Examples

To set the Fibre Channel port speed:

```
device (config)# interface FibreChannel 7/0/2
device(conf-FibreChannel-7/0/2)# speed 4gbps
```

# speed (FlexPort)

This command sets the protocol and speed for the FlexPort connector group.

## Syntax

```
speed { LowMixed | HighMixed | FibreChannel}
```

## Command Default

The default state is Ethernet.

## Parameters

### LowMixed

Sets to speed to 2/4/8G Fibre Channel and Ethernet speeds.

### HighMixed

Sets the speed to 16G Fibre Channel and Ethernet speeds

### FibreChannel

Sets the speed to support only fibre channel speeds and protocol. All FlexPorts in this connector-group must be converted to fibre-channel in order to use the FibreChannel connector-group speed.

## Modes

Hardware connector-group configuration mode.

## Usage Guidelines

Changing connector-group speed is disruptive to all ports within the group depending on their configuration and the new connector-group to be used. The user will be warned about potential disruption. A speed change will be disallowed if any port in the connector group is already running at a speed that cannot be supported with the new connector group speed.

In order to change to the 2/4/8/16G connector-group speed, all ports within the connector-group first need to be configured to the Fibre Channel port type.

## Examples

This example configures the speed for the FlexPort on Rbridge-ID 47, connector group 6, to support Fibre Channel.

```
switch(conf-if-fi-47/0/8)#speed FibreChannel
```

## History

Release version	Command history
5.0.0	This command was introduced.



## Related Commands

connector-group, flexport, hardware

# speed (LAG)

Sets the speed on a LAG interface.

## Syntax

```
speed { 1000 | 10000 | 40000 }
```

## Command Default

Speed is 10000

## Parameters

**1000**

Forces the speed to 1 Gbps.

**10000**

Forces the speed to 10 Gbps.

**40000**

Forces the speed to 40 Gbps.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The speed command is available only for 10-gigabit Ethernet ports.

# speed (port-channel)

Sets the speed on a port-channel interface.

## Syntax

```
speed { 1000 | 10000 | 40000 | 100000 }  
no speed
```

## Command Default

Speed is 10000.

## Parameters

**1000**  
Forces the speed to 1 Gbps.

**10000**  
Forces the speed to 10 Gbps.

**40000**  
Forces the speed to 40 Gbps.

**100000**  
Forces the speed to 100 Gbps. This is available only if the HundredGigabit line card is supported.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no speed** to return to the default setting.

## Examples

To set the speed to 40 Gbps on a specific port-channel interface:

```
switch(config)# interface port-channel 44  
switch(config-Port-Channel-44)# speed 40000
```

## Related Commands

[interface](#)

# spt-threshold

Configures the Shortest Path Tree (SPT) threshold.

## Syntax

```
spt-threshold { infinity | num }
```

```
no spt-threshold
```

## Command Default

Default value is 1.

## Parameters

**infinity**

Use only the rendezvous point to send packets, do not switch over to SPT.

*num*

Rate (in kilobytes per second) that must be reached before switching to SPT. Valid values range from 1 through 4294967295.

## Modes

PIM router configuration mode

## Usage Guidelines

This command sets the rate, in kilobytes per second, data is to be sent through the rendezvous point before switching to SPT for sending packets.

Enter **no spt-threshold** to return to the default setting of 1.

## Examples

To set the SPT threshold interval to 20:

```
switch(conf-pim-router) # spt-threshold 20
```

## Related Commands

[router pim](#)

# ssh

Connects to a remote server by means of the Secure Shell (SSH) protocol.

## Syntax

```
ssh { IP_address | hostname } [ -c | -l | -m | interface { ethernet slot/port | management | ve vlan-id } | vrf vrf-name ] }
```

## Command Default

SSH connects to port 22.

## Parameters

*IP\_address*

Specifies the server IP address in IPv4 or IPv6 format.

*hostname*

Specifies the host name, a string from 1 through 253 characters.

**-c**

Specifies the encryption algorithm for the SSH session. This parameter is optional. Supported algorithms include the following:

**aes128-cbc**

AES 128-bits

**aes192-cbc**

AES 192-bits

**aes256-cbc**

AES 256-bits

**-l *username***

Login name for the remote server. This parameter is optional. If you specify a user name, you will be prompted for a password. If you do not specify a user name, the command assumes you are logging in as root and will prompt for the root password.

**-m**

Specifies the HMAC (Hash-based Message Authentication Code) message encryption algorithm. This parameter is optional; if no encryption algorithm is specified, the default (**hmac-md5**) is used. Supported algorithms include the following:

**hmac-md5**

MD5 128-bits. This is the default setting.

**hmac-md5-96**

MD5 96-bits

**hmac-sha1**

SHA1 160-bits

**hmac-sha1-96**

SHA1 96-bits

**interface**

Specifies an interface.

**ethernet** *slot/port*

Specifies an Ethernet interface slot and port number. The valid value is 0.

**management**

Specifies a management interface.

**ve** *vlan-id*

Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

**vrf** *vrf-name*

Specifies a VRF instance. See the Usage Guidelines.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to establish an encrypted SSH connection from a switch to a remote networking device. This implementation is based on SSH v2.

To use the **ssh** command on the management VRF, use the **vrf** keyword and enter **mgmt-vrf** manually.

The following features are not supported:

- Displaying SSH sessions
- Deleting stale SSH keys

## Examples

To connect to a remote device using an SSH connection with default settings:

```
device# ssh 10.70.212.152
```

```
The authenticity of host '10.70.212.152 (10.70.212.152)' can't be established.
RSA key fingerprint is f0:2a:7e:48:60:cd:06:3d:f4:44:30:2a:ce:68:fe:1d.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.70.212.152' (RSA) to the list of known hosts.
Password:
```

To connect to a remote device using an SSH connection with the management VRF:

```
device# ssh 10.70.212.152 vrf mgmt-vrf
```

To connect to a remote device using an SSH connection with a login name:

```
device# ssh -l admin 127.2.1.8
```

```
admin@127.2.1.8's password
```

## Related Commands

[show running-config ssh server](#), [ssh server shutdown](#), [telnet](#)

# ssh client cipher

Sets the SSH client's cipher list for the SSH client.

## Syntax

`ssh client cipher string`

`no ssh client cipher`

## Parameters

*string*

The string name of the cipher. Refer to the device for the available options.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the `no ssh client cipher` command remove the cipher list from the ssh client.

## Examples

Sets the SSH client's cipher list.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ssh client cipher aes128-cbc
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ssh client cipher non-cbc

Sets the SSH client's cipher list to non-cbc ciphers for the SSH client.

## Syntax

`ssh client cipher non-cbc`

`no ssh client ciphe non-cbcr`

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the `no ssh client cipher non-cbc` command remove the non-cbc cipher list from the ssh client.

## Examples

Sets the SSH client's cipher list to non-cbc ciphers.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ssh client cipher non-cbc
device(config-rbridge-id-1)# do show running-config rbridge-id ssh
rbridge-id 1
ssh server non-cbc
ssh client non-cbc
```

## History

Release version	Command history
5.0.1	This command was introduced.
6.0.0	This command was enhanced.

## Related Commands

[ssh server cipher non-cbc](#)



# ssh client key-exchange

Specifies the method used for generating the one-time session keys for encryption and authentication with the Secure Shell (SSH) server and Diffie-Hellman group 14.

## Syntax

```
ssh client key-exchange string
```

```
no ssh client key-exchange
```

## Parameters

*string*

The string for the name of the algorithm `diffie-hellman-group14-sha1`, or a comma-separated list of supported Key-exchange algorithms; such as `diffie-hellman-group14-sha1,diffie-hellman-group1-sha1`, and so on.

## Command Default

This command is not configured by default.

## Modes

RBridge ID configuration mode

## Usage Guidelines

You can configure the SSH client key-exchange method to DH Group 14. When the ssh client key-exchange method is configured to DH Group 14, the SSH connection from a remote SSH client is allowed only if the key-exchange method at the client end is also configured to DH Group 14. Enter **no ssh client key-exchange** to restore ssh client key-exchange to the default value.

This command is not distributed across the cluster. The RBridge ID of the node should be used to configure service on individual nodes.

For information on DH Group 14, refer to [RFC 3526](#).

For backward compatibility, the string "dh-group-14" is also acceptable in place of "diffie-hellman-group14-sha1"

## Examples

To set ssh client key-exchange to DH Group 14:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# ssh client key-exchange diffie-hellman-group14-sha1
```

To restore the ssh client key-exchange to default value:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# no ssh client key-exchange
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ssh client mac

Supports MAC configurations for the SSH client.

## Syntax

`ssh client mac string`

`no ssh client mac`

## Command Default

SSH server is enabled by default.

## Parameters

*string*

The string name of the default MAC required. Your choices are hmac-md5, hmac-sha1, hmac-sha2-256, and hmac-sha2-512. The default MACs supported in FIPS mode are hmac-sha1, hmac-sha2-256, and hmac-sha2-512.

## Modes

RBridge ID configuration mode

## Usage Guidelines

The MAC hmac-md5 is not supported in FIPS mode.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-176)# ssh client mac hmac-sha1,hmac-sha2-256,hmac-sha2-512
device(config-rbridge-id-176)# do show running-config rbridge-id ssh client
rbridge-id 176
ssh client mac hmac-sha1,hmac-sha2-256,hmac-sha2-512
!
device(config-rbridge-id-176)# do show ssh client status rbridge-id 176
rbridge-id 176:SSH Client Mac: hmac-sha1,hmac-sha2-256,hmac-sha2-512
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ssh server cipher

Sets the SSH server's cipher list for the SSH server.

## Syntax

**ssh server cipher** *string*

**no ssh server cipher**

## Parameters

*string*

The string name of the cipher. Refer to the device for the available options.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no ssh server cipher** command remove the cipher list from the ssh client.

## Examples

Sets the SSH server's cipher list.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ssh server cipher aes256-ctr
```

## History

Release version	Command history
6.0.1	This command was introduced.

# ssh server cipher non-cbc

Sets the SSH server's cipher list to non-cbc ciphers for the SSH server.

## Syntax

```
ssh server cipher non-cbc
```

```
no ssh server cipher non-cbc
```

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no ssh server cipher non-cbc** command remove the non-cbc cipher list from the ssh client.

## Examples

Sets the SSH server's cipher list to non-cbc ciphers.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# ssh server cipher non-cbc
device(config-rbridge-id-1)# do show running-config rbridge-id ssh
rbridge-id 1
ssh server non-cbc
ssh client non-cbc
switch(config-rbridge-id-1)#
```

## History

Release version	Command history
5.0.1	This command was introduced.
6.0.0	This command was enhanced.

## Related Commands

[ssh client cipher non-cbc](#)

# ssh server key-exchange

Specifies the method used for generating the one-time session keys for encryption and authentication with the Secure Shell (SSH) server and Diffie-Hellman group 14.

## Syntax

```
ssh server key-exchange string
```

```
no ssh server key-exchange
```

## Parameters

*string*

The string for the name of the algorithm `diffie-hellman-group14-sha1`, or a comma-separated list of supported Key-exchange algorithms; such as `diffie-hellman-group14-sha1,diffie-hellman-group1-sha1`, and so on.

## Command Default

This command is not configured by default.

## Modes

RBridge ID configuration mode

Global configuration mode

## Usage Guidelines

You can configure the SSH server key-exchange method to DH Group 14. When the SSH server key-exchange method is configured to DH Group 14, the SSH connection from a remote SSH client is allowed only if the key-exchange method at the client end is also configured to DH Group 14. Enter **no ssh server key-exchange** to restore SSH server key-exchange to the default value.

This command is not distributed across the cluster. The RBridge ID of the node should be used to configure service on individual nodes.

For information on DH Group 14, refer to [RFC 3526](#).

For backward compatibility, the string "dh-group-14" is also acceptable in place of "diffie-hellman-group14-sha1"

## Examples

To set SSH server key-exchange to DH Group 14:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# ssh server key-exchange diffie-hellman-group14-sha1
```

To restore the SSH server key-exchange to default value:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# no ssh server key-exchange
```

## Related Commands

[show running-config ssh server](#), [show ssh server status](#)

# ssh server mac

Supports MAC configurations for the SSH server.

## Syntax

**ssh server mac** *string*

**no ssh server mac**

## Parameters

*string*

The string name of the default MAC required. Your choices are hmac-md5, hmac-sha1, hmac-sha2-256, and hmac-sha2-512. The default MACs supported in FIPS mode are hmac-sha1, hmac-sha2-256, and hmac-sha2-512.

## Modes

RBridge ID configuration mode

## Usage Guidelines

The MAC hmac-md5 is not supported in FIPS mode.

## Examples

Typical command example:

```
device# configure terminal
device(config)# rbridge-id 1
device(config)#rbridge-id 176
device(config-rbridge-id-176)# ssh server mac hmac-sha1,hmac-sha2-256,hmac-sha2-512
device(config-rbridge-id-176)# do show running-config rbridge-id ssh server
rbridge-id 176
ssh server mac hmac-sha1,hmac-sha2-256,hmac-sha2-512
ssh server key rsa 2048
ssh server key ecdsa 256
ssh server key dsa
```

## History

Release version	Command history
6.0.1	This command was introduced.



# ssh server max-sessions

Specifies the maximum number of open Secure Shell (SSH) sessions per SSH network connection.

## Syntax

**ssh server max-sessions** *number*

**no ssh server max-sessions**

## Command Default

The default number of sessions is 1 unless it is changed by this command.

## Parameters

*number*

Maximum number of sessions. Range is from 1 through 10.

## Modes

RBridge ID configuration mode

## Usage Guidelines

After executing this command, in order to use the new number of sessions, you must first shut down the SSH server, by means of the **ssh server use-vrf shutdown** command, and then restart it, by means of the **no ssh server use-vrf shutdown** command.

The maximum number of sessions specified by this command is synchronized to the standby management module (MM). However, to make the change effective on the standby MM, you must first disable service on that module by means of the **no ssh server standby enable** command, and then reenables service by means of the **ssh server standby enable** command.

Use the **show running-config rbridge-id ssh server** command or the **show ssh server status** command to confirm the configuration.

A downgrade to a previous release is blocked if this command has been executed in the running configuration.

Use the **no ssh server max-sessions** command to revert to the default of 1 session. You must also stop and restart service as in the Usage Guidelines above.

## Examples

To change the maximum number of supported SSH sessions from the default to 7, and confirm the configuration:

```
device# configure terminal
device(config)# rbridge-id 176
device(config-rbridge-id-176)# ssh server max-sessions 7
device(config-rbridge-id-176)# do show running-config rbridge-id ssh server
rbridge-id 176
ssh server max-sessions 7
ssh server key rsa 2048
ssh server key ecdsa 256
ssh server key dsa
```

To revert to the default number of sessions (1):

```
device(config-rbridge-id-176)# no ssh server max-sessions
```

## History

Release version	Command history
7.1.0	This command was introduced.

# ssh server rekey-interval

Configures the Secure Shell (SSH) server rekey-interval.

## Syntax

```
ssh server rekey-interval interval
```

```
no ssh server rekey-interval
```

## Parameters

*interval*

The value for the rekey interval. Range is from 900 to 3600 seconds.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use the **no ssh server rekey-interval** command to remove the rekey-interval.

# ssh server shutdown

Disables SSH service.

## Syntax

```
ssh server [ use-vrf vrf-name ] shutdown
```

```
no ssh server [ use-vrf vrf-name ] shutdown
```

## Parameters

**use-vrf** *vrf-name*

Specifies a user-defined VRF, or built-in VRFs such as mgmt-vrf or default-vrf.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Enter **no ssh server shutdown** to enable SSH service.

The **no ssh server shutdown** command is not distributed across the cluster. The RBridge ID of the node should be used to configure service on individual nodes.

The use of the **use-vrf** keyword brings down the server only for the specified VRF. The user can shut down any server in any VRF, including the management and default VRF.

When this command is executed and a VRF is not specified by means of the **use-vrf** keyword, the server is brought down only in the management VRF ("mgmt-vrf") (the default VRF for this command).

When this command is executed at the RBridge ID level for a specified VRF, connectivity to servers in that VRF is enabled, whereas this service for nonspecified RBridges is shut down.

## Examples

To shut down SSH service on the management VRF:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# ssh server shutdown
```

To shut down SSH service for a user-defined VRF:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# ssh server use-vrf myvrf shutdown
```

To enable SSH service on the management VRF:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# no ssh server shutdown
```

To enable SSH service:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# no ssh server shutdown
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[show support](#), [telnet server shutdown](#), [show running-config ssh server](#), [show ssh server status](#)

# ssh server standby enable

Enables the SSH services on the standby MM.

## Syntax

**ssh server standby enable**

**no ssh server standby enable**

## Command Default

The SSH services are disabled on the standby MM.

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no ssh server standby enable** command disables the SSH services on the standby MM.

## Examples

Typical command output:

```
device(config-rbridge-id-1)# no ssh server standby enable
device(config-rbridge-id-1)# do show running-config rbridge-id | include standby
% No entries found.
```

## History

Release version	Command history
5.0.1a	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[telnet server standby enable](#)

# static-network

Configures a static BGP4 network, creating a stable network in the core.

## Syntax

**static-network** *network/mask* [ **distance** *num* ]

**no static-network** *network/mask* [ **distance** *num* ]

## Parameters

*network/mask*

Network and mask in CIDR notation.

**distance** *num*

Specifies an administrative distance value for this network. Valid values range from 1 through 255. The default is 200.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

## Usage Guidelines

While a route configured with this command will never flap unless it is deleted manually, a static BGP4 network will not interrupt the normal BGP4 decision process on other learned routes that are installed in the Routing Table Manager (RTM). Consequently, when there is a route that can be resolved, it will be installed into the RTM.

## Examples

The following example configures a static network and sets an administrative distance of 300.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# static-network 10.11.12.0/32 distance 300
```

The following example configures a static network for VRF instance "red" and sets an administrative distance of 300.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# static-network 10.11.12.0/24 distance 300
```

## History

Release version	Command history
6.0.1	Support was added for the BGP address-family IPv4 unicast VRF configuration mode.

## Related Commands

[route-map](#)



# storm-control ingress

Limits ingress traffic on a specified interface.

## Syntax

```
storm-control ingress { broadcast | unknown-unicast | multicast } { limit-bps | limit-percent } rate [ { monitor | shutdown } ]
no storm-control ingress { broadcast | unknown-unicast | multicast } { limit-bps | limit-percent } rate [ { monitor | shutdown } ]
```

## Parameters

### broadcast

Specifies that the command will operate on broadcast traffic only.

### unknown-unicast

Specifies that the command will operate on unknown-unicast traffic only.

### multicast

Specifies that the command will operate on multicast traffic only.

### limit-bps

Specifies that the value given to the *rate* parameter is in bits per second. If the traffic on the interface reaches this rate, no more traffic (for the traffic type specified) is allowed on the interface.

### limit-percent

Specifies that the value given to the *rate* parameter is in percentage of capacity of the interface. If the traffic on the interface reaches this percentage of capacity, no more traffic (for the traffic type specified) is allowed on the interface.

### *rate*

Specifies the amount of traffic allowed, either in bits per second or a percentage of the capacity of the interface, depending on which parameter was chosen with the rate.

If you are specifying rate in bps, enter an integer from 0 to 10000000000. Because each application-specific integrated circuit (ASIC) may support different bit granularity, bit rates are rounded up to the next achievable rate.

If you are specifying rate in percent of interface capacity, enter an integer from 0 to 100.

### monitor

Specifies that, if a rate limit is reached within a five-second sampling period, a log message gets sent. A log message is generated upon the first occurrence of such an event. Subsequent log messages are generated only at the end of one complete sample interval in which no rate limits are reached.

### shutdown

Specifies that, if a rate limit is exceeded within a five-second sampling period, the interface will be shut down. You must manually re-enable the interface after a shutdown.

## Modes

Interface configuration mode

## Usage Guidelines

This command limits the amount of broadcast, unknown unicast, and multicast (BUM) ingress traffic on a specified interface. The *shutdown* parameter monitors the status of the configured rate limit every five seconds, and if the maximum defined rate is exceeded the corresponding interface is shut down until you re-enable it using the **no shut** command.

This command is supported on the VDX 6740 6740, VDX 8770-4 and VDX 8770-8 platforms only.

If you want to modify an active BUM storm control configuration, you must first disable it, then issue the **storm-control ingress** command again with the new parameters.

Enter **no storm-control ingress** to disable BUM storm control for a particular traffic type on an interface.

## Examples

To configure storm control on a 10-gigabit Ethernet interface, with a rate limited to 1000000 bps:

```
device(config)# interface tengigabitethernet 101/0/2
device(conf-if-te-101/0/2)# storm-control ingress broadcast 1000000
```

## Related Commands

[clear counters storm-control](#), [interface](#), [show storm-control](#)

# summary-address (OSPFv2)

Configures route summarization for redistributed routes for an Autonomous System Boundary Router (ASBR).

## Syntax

```
summary-address A.B.C.D E.F.G.H  
no summary-address
```

## Command Default

Summary addresses are not configured.

## Parameters

*A.B.C.D E.F.G.H*  
IP address and mask for the summary route representing all the redistributed routes in dotted decimal format.

## Modes

OSPF router configuration mode  
OSPF VRF router configuration mode

## Usage Guidelines

Use this command to configure an ASBR to advertise one external route as an aggregate for all redistributed routes that are covered by a specified address range. When you configure an address range, the range takes effect immediately. All the imported routes are summarized according to the configured address range. Imported routes that have already been advertised and that fall within the range are flushed out of the AS and a single route corresponding to the range is advertised.

If a route that falls within a configured address range is imported by the device, no action is taken if the device has already advertised the aggregate route; otherwise the device advertises the aggregate route. If an imported route that falls within a configured address range is removed by the device, no action is taken if there are other imported routes that fall within the same address range; otherwise the aggregate route is flushed.

You can configure up to 32 address ranges.

The device sets the forwarding address of the aggregate route to 0 and sets the tag to 0. If you delete an address range, the advertised aggregate route is flushed and all imported routes that fall within the range are advertised individually. If an external link-state-database-overflow condition occurs, all aggregate routes and other external routes are flushed out of the AS. When the device exits the external LSDB overflow condition, all the imported routes are summarized according to the configured address ranges. This parameter affects only imported, type 5 external routes.

The no form of the command disables route summarization.

## Examples

The following example configures a summary address of 10.1.0.0 with a mask of 10.255.0.0. Summary address 10.1.0.0, includes addresses 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. For all of these networks, only the address 10.1.0.0 is advertised in external LSAs:

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# summary-address 10.1.0.0 10.255.0.0
```

# summary-address (OSPFv3)

Configures route summarization for redistributed routes for an Autonomous System Boundary Router (ASBR).

## Syntax

```
summary-address IPv6-addr/mask  
no summary-address
```

## Command Default

Summary addresses are not configured.

## Parameters

*A:B:C:D/LEN*

IPv6 address and mask for the summary route representing all the redistributed routes in dotted decimal format.

## Modes

OSPFv3 router configuration mode  
OSPFv3 VRF router configuration mode

## Usage Guidelines

Use this command to configure an ASBR to advertise one external route as an aggregate for all redistributed routes that are covered by a specified IPv6 address range. When you configure an address range, the range takes effect immediately. All the imported routes are summarized according to the configured address range. Imported routes that have already been advertised and that fall within the range are flushed out of the AS and a single route corresponding to the range is advertised.

If a route that falls within a configured address range is imported by the device, no action is taken if the device has already advertised the aggregate route; otherwise the device advertises the aggregate route. If an imported route that falls within a configured address range is removed by the device, no action is taken if there are other imported routes that fall within the same address range; otherwise the aggregate route is flushed.

You can configure up to 4 address ranges.

The device sets the forwarding address of the aggregate route to 0 and sets the tag to 0. If you delete an address range, the advertised aggregate route is flushed and all imported routes that fall within the range are advertised individually. If an external link-state-database-overflow condition occurs, all aggregate routes and other external routes are flushed out of the AS. When the device exits the external LSDB overflow condition, all the imported routes are summarized according to the configured address ranges.

## Examples

The following example configures a summary address of 2001:db8::/24 for routes redistributed into OSPFv3. The summary prefix 2001:db8::/24 includes addresses 2001:db8::/1 through 2001:db8::/24. Only the address 2001:db8::/24 is advertised in an external link-state advertisement.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# summary-address 2001:db8::/24
```

# support autoupload enable

Specifies if support autoupload is enabled or disabled. When set to enabled, the data files are automatically transferred to the configured remote location.

## Syntax

```
support autoupload enable  
no support autoupload enable
```

## Command Default

Support autoupload is disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

Whenever a core file, FFDC, trace data file occurs, the data files are automatically transferred to the configured remote location if the autoupload feature is enabled.

Use the **no** form of this command to disable support autoupload.

## Examples

To enable autoupload mode:

```
switch(config)# support autoupload enable
```

To disable autoupload mode:

```
switch(config)# no support autoupload enable
```

## Related Commands

[show running-config support autoupload-param](#)

# support autoupload-param

Defines autoupload parameters.

## Syntax

```
support autoupload-param hostip host-ip user user_acct password password protocol [ ftp | scp | sftp ] directory path
```

## Parameters

**hostip** *host-ip*

Specifies the IP address of the remote host.

**user** *user\_acct*

Specifies the user name to access the remote host.

**password** *password*

Specifies the password to access the remote host.

**protocol** FTP | SCP | SFTP

Specifies the protocol used to access the remote server.

**directory** *path*

Specifies the path to the directory.

**rbridge-id**

Enables RBridge ID mode to support Virtual Cluster Switching (VCS) on individual nodes.

*rbridge-id*

Specifies a unique identifier for a node.

**all**

Specifies all identifiers for a node.

## Modes

Global configuration mode

## Examples

To configure autoupload parameters:

```
switch(config)# support autoupload-param hostip 10.31.2.27 protocol [ftp|scp | sftp]username hegdes
directory /uers/home40/hegdes/autoupload password
(<string>): *****
```



# support support-param

Defines support parameters.

## Syntax

```
support support-param hostip host-ip user user_acct password password protocol [ ftp | scp | sftp ] directory path
```

## Parameters

**hostip** *host-ip*

Specifies the IP address of the remote host.

**user** *user\_acct*

Specifies the user name to access the remote host.

**password** *password*

Specifies the password to access the remote host.

**protocol** FTP | SCP | SFTP

Specifies the protocol used to access the remote server.

**directory** *path*

Specifies the path to the directory.

## Modes

Global configuration mode

## Examples

To configure support parameters:

```
switch(config)# support support-param hostip 10.31.2.27 protocol [ftp|scp | sftp]username hegdes
directory /uers/home40/hegdes/support password
```

```
(<string>): *****
```

# suppress-arp

Enables Address Resolution Protocol (ARP) suppression on a current VLAN. ARP suppression can lessen ARP-related traffic within an IP Fabric.

## Syntax

```
suppress-arp
```

```
no suppress-arp
```

## Command Default

ARP suppression is disabled.

## Modes

VLAN configuration mode

## Usage Guidelines

This feature is required, along with ND suppression, if static anycast gateway is supported in an IP Fabric.

To disable ARP suppression, use the **no** form of this command.

## Examples

The following example enables ARP suppression on VLAN 110.

```
device# configure terminal
device(config)# interface vlan 110
device(config-Vlan-110)# suppress-arp
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.0.1	The Usage Guidelines were updated.

# suppress-nd

Enables Neighbor Discovery (ND) suppression on a VLAN. ND suppression can lessen the amount of ND control traffic within an IP Fabric.

## Syntax

```
suppress-nd
no suppress-nd
```

## Command Default

ND suppression is disabled.

## Modes

VLAN configuration mode

## Usage Guidelines

This command can be applied to a maximum of 512 VLANs.

This feature is required, along with ARP suppression, if static anycast gateway is supported in an IP Fabric.

To disable ND suppression, use the **no** form of this command.

## Examples

The following example enables ND suppression on VLAN 110.

```
device# configure terminal
device(config)# interface vlan 110
device(config-vlan-110)# suppress-nd
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.0.1	The Usage Guidelines were updated.

# switch-attributes

Sets switch attributes.

## Syntax

**switch-attributes** *rbridge-id*

**chassis-name** *string*

**host-name** *string*

**no switch-attributes**

## Command Default

The default chassis name depends on the switch model. You can assign the chassis name any name you wish to represent one of the following product names:

- VDX 6740
- VDX 6740T
- VDX 67440T-1G
- VDX 8770-4
- VDX 8770-8

The default host name is "sw0".

## Parameters

*rbridge-id*

Specifies the RBridge ID the attribute is to be set for. Only the local RBridge ID is supported.

**chassis-name** *string*

Sets the switch chassis name. The string must be between 1 and 30 ASCII characters in length, and the leading character must be a letter.

**host-name** *string*

Sets the switch host name. The string must be between 1 and 30 ASCII characters in length, and the leading character must be a letter.

## Modes

Global configuration mode

## Usage Guidelines

When issued with the RBridge ID of the switch to be configured, this command goes into a sub-command shell where you can configure the host name or chassis name.

The text string for the **chassis-name** and **host-name** string is limited to 30 characters. The string must begin with a letter, and can consist of letters, digits, hyphens, periods (dots), and underscore characters. Spaces are not permitted.

This command is not supported on the standby management module.

This command is supported only on the local switch.

Enter **no switch-attributes** to restore the default values.

## Examples

To set the host name for a switch with an RBridge ID of 2:

```
switch(config)# switch-attributes 2
switch(config-switch-attributes-1)# host-name VDX8770-4
```

## Related Commands

[show running-config switch-attributes](#)

# switchport

Puts the interface in Layer 2 mode and sets the switching characteristics of the Layer 2 interface.

## Syntax

**switchport**

**no switchport**

## Command Default

All Layer 2 interfaces are mapped to default VLAN 1 and the interface is set to access mode.

## Modes

Interface subtype configuration mode

## Usage Guidelines

For changing the interface configuration mode to trunk or changing the default VLAN mapping, use additional **switchport** commands.

To redefine the switch from Layer 2 mode into Layer 3 mode, enter **no switchport**.

## Examples

To put a specific 10-gigabit Ethernet interface in Layer 2 mode:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# switchport
```

To remove a specific port-channel interface from Layer 2 mode:

```
switch(config)# interface port-channel 44
switch(config-port-channel-44)# no switchport
```

# switchport access

Sets the Layer 2 interface as access.

## Syntax

```
switchport access { vlan vlan_id | rspan-vlan vlan_id | mac HHHH.HHHH.HHHH | mac-group mac-group-id }
no switchport access { vlan vlan_id | rspan-vlan vlan_id | mac HHHH.HHHH.HHHH | mac-group mac-group-id }
```

## Command Default

All Layer 2 interfaces are in access mode and belong to the VLAN ID 1.

## Parameters

**vlan** *vlan\_id*

Sets the port VLAN (PVID) to the specified *vlan\_id*. Range is below 4096 for 802.1Q VLANs, and from 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.

**rspan-vlan** *vlan\_id*

Sets a VLAN ID for RSPAN (Remote Switched Port Analyzer) traffic analysis.

**mac** *HHHH.HHHH.HHHH*

Sets a source MAC address for classifying an untagged VLAN specified by the **vlan** keyword.

**mac-group** *mac-group-id*

(Optional) Specifies a set of MAC addresses. The group of addresses must be established by the global **mac-group** command.

## Modes

Interface subtype configuration mode on edge ports

## Usage Guidelines

In access mode, the interface only allows untagged and priority tagged packets.

In a Virtual Fabrics context, use this command also to configure service or transport VFs on an access port. This allows multiple untagged VLANs on the port by means of SRC MAC classifiers.

Enter **no switchport access vlan** to set the PVID to the default VLAN 1.

## Examples

To set the Layer 2 interface PVID to 100 on a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# switchport access vlan 100
```

To set the PVID to the default VLAN 1 on a specific port-channel interface:

```
switch(config)# interface port-channel 44
switch(config-port-channel-44)# no switchport access vlan
```

The following examples illustrate configuration with service or transport VFs in a Virtual Fabrics context.

In global configuration mode, establish a mac-group:

```
switch(config)# mac-group 1
switch(config-mac-group 1)# mac 0002.0002.0002
switch(config-mac-group 1)# mac 0005.0005.0005
switch(config-mac-group 1)# mac 0008.0008.0008
```

In interface configuration mode, ensure that the switchport mode is set to access:

```
switch(config)# int te 2/0/1
switch(config-if-te-2/0/1)# switchport mode access
```

Set the default access VLAN (the default is 1) to 5000 (a classified VLAN):

```
switch(config-if-te-2/0/1)# switchport access vlan 5000
```

Classify an 802.1Q VLAN by means of a source MAC address:

```
switch(config-if-te-2/0/1)# switchport access vlan 200 mac 0002.0002.0002
```

Configure a classified VLAN (> 4095) on the same interface with a MAC address. Frames that do not match the source MAC addresses of 0002.0002.0002 or 0004.0004.0004 are classified into VLAN 5000 (the access VLAN for all untagged frames that do not have MAC address classifications).

```
switch(config-if-te-2/0/1)# switchport access vlan 6000 mac 0004.0004.0004
```

The following errors occur because a MAC address can be classified to only one VLAN on the same interface.

```
switch(config-if-te-2/0/1)# switchport access vlan 7000 mac-group 1
switch(config-if-te-2/0/1)# %Error: Mac-address/Mac-group is overlapping with another Mac-address/Mac-group configuration on the same port.
switch(config-if-te-3/0/1)# switchport mode access
switch(config-if-te-3/0/1)# switchport access vlan 7000 mac-group 1
switch(config-if-te-3/0/1)# switchport access vlan mac 8000 0008.0008.0008
switch(config-if-te-3/0/1)# %Error: Mac-address/Mac-group is overlapping with another Mac-address/Mac-group configuration on the same port.
```



# switchport mode

Sets the mode of the Layer 2 interface.

## Syntax

```
switchport mode { access | trunk }
```

## Parameters

### access

Sets the Layer 2 interface as access. Access mode assigns the port to a VLAN

### trunk

Sets the Layer 2 interface as trunk. Trunk mode makes the port linkable to other switches and routers

## Modes

Interface subtype configuration mode

## Usage Guidelines

You must configure the same native VLAN on both ends of an 802.1 or classified VLAN trunk link. Failure to do so can cause bridging loops and VLAN leaks.

## Examples

To set the mode of a specific 10-gigabit Ethernet interface to *access* :

```
switch(config)# interface tengigabitethernet 178/0/9  
switch(conf-if-te-178/0/9)# switchport mode access
```

To set the mode of a specific port-channel interface to *trunk*:

```
switch(config)# interface port-channel 44  
switch(config-port-channel-44)# switchport mode trunk
```

# switchport mode private-vlan

Sets the private VLAN (PVLAN) mode of the Layer 2 interface.

## Syntax

```
switchport mode private-vlan [ host ] [ promiscuous ] [ trunk [ promiscuous | host ] ]
```

## Command Default

The port does not have any PVLAN attributes by default.

## Parameters

### host

Sets the port mode to host (community or isolated) mode. It accepts the untagged or priority tagged packet, and the outgoing packet is untagged.

### promiscuous

Sets the port mode to promiscuous mode.

### trunk

Sets the port mode to PVLAN trunk port. This port can carry multiple VLANs. The outgoing packets carry all VLANs, except for native VLANs.

### trunk host

Sets the port mode to host (community or isolated) mode. The trunk operand means the outgoing packet will be tagged "accept".

### trunk promiscuous

Sets the trunk to promiscuous mode.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command assigns the primary Vlan to a promiscuous port. This command also maps a promiscuous port to selected secondary VLANs. This means only selected VLANs can send packets to this port.

All switchport modes are independent from each other, including normal mode (access/trunk) and above private VLAN modes. Based on the default behavior of the port, the new mode automatically overwrites the existing mode by deleting the existing mode (removing any relationship/association) and applying the new mode.

## Examples

To set the mode of a specific 10-gigabit Ethernet interface to PVLAN trunk:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# switchport mode private-vlan trunk
```

To set the mode of a specific 10-gigabit Ethernet interface to PVLAN promiscuous (untagged):

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# switchport mode private-vlan promiscuous
```

To set the mode of a specific 10-gigabit Ethernet interface to PVLAN promiscuous (tagged):

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# switchport mode private-vlan trunk promiscuous
```

# switchport mode trunk-no-default-native

Configures a port to trunk mode without the implicit creation of default native VLAN 1 in a Virtual Fabrics context.

## Syntax

```
switchport mode trunk-no-default-native
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

When this command is enabled, any ingress tagged or untagged packet is discarded until a switchport classification or native VLAN classification is configured. To disable this functionality, simply issue the **no switchport** command, or enter a different switchport mode by using the **switchport mode access** command or the **switchport mode trunk** command.

Port mode change is not allowed when port security is enabled on the interface.

This is the fundamental difference between this command and the **switch mode trunk** command, which implicitly creates VLAN 1 on the port.

The global command **dot1q tag native-vlan** does not affect the ingress or egress tagging behavior of the native VLAN configured in this mode.

The following native VLAN commands are supported in this mode:

- **switchport trunk native-vlan-untagged**
- **switchport trunk native-vlan-xtagged**

The following native VLAN commands that are supported in regular trunk mode are NOT supported in this mode:

- **switchport trunk tag native-vlan**
- **switchport trunk native-vlan**

## Examples

Configure a trunk port without a default native VLAN, then explicitly configure the native VLAN.

```
switch(config)# interface te 2/1/1
switch(config-if-te-2/1/1)# switchport mode trunk-no-default-native
switch(config-if-te-2/1/1)# switchport trunk native-vlan-xtagged 1 egress tagged
```

## Related Commands

[switchport trunk native-vlan](#), [switchport trunk native-vlan-untagged](#), [switchport trunk native-vlan-xtagged](#)

# switchport port-security

Enables port security on an interface port.

## Syntax

**switchport port-security**

**no switchport port-security**

## Command Default

Port security is not enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Port mode change is not allowed when port security is enabled on the interface.

The **no switchport port-security** command disables port security on the interface.

## Examples

The following example enables port MAC security on an interface:

```
device(config)# interface TenGigabitEthernet 1/0
device(conf-if-te-1/0)# switchport
device(conf-if-te-1/0)# switchport port-security
```

# switchport port-security mac-address

Configures the MAC address option for port security on an interface port.

## Syntax

```
switchport port-security mac-address address vlan vlan_id
```

## Command Default

MAC address is not configured for port security.

## Parameters

**mac-address** *address*

Specifies the MAC address-based VLAN classifier rule used to map to a specific VLAN.

**vlan** *vlan\_id*

Specifies a VLAN.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Static MAC addresses cannot be configured on a secure port. They must be configured as secure MAC addresses on the secure port.

When static MAC address is configured on an access secure port, the MACs qualify for access VLANs, but on trunk port, VLAN must be specified.

The **no switchport port-security mac-address** command removes the specified MAC address.

## Examples

The following example configures static MAC address for port security on an interface:

```
device(config)# interface TenGigabitEthernet 1/0
device(conf-if-te-1/0)# switchport
device(conf-if-te-1/0)# switchport port-security mac-address 1000.2000.3000 vlan 100
```

# switchport port-security max

Configures the maximum number of MAC addresses used for port MAC security on an interface port.

## Syntax

```
switchport port-security max value  
no switchport port-security max
```

## Parameters

*value*

The maximum number of secure MAC addresses. Range is from 1 through 8192.

## Command Default

The default value is 8192 MAC addresses.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The maximum MAC address limit for sticky MAC address and static MAC address depends on the device limit. For dynamically learned MAC addresses, the maximum limit is 8192 per port.

The **no switchport port-security max** command restores the default value of maximum number of MAC addresses.

## Examples

The following example configures the maximum number of MAC addresses used for port MAC security on an interface port as 10:

```
device(config)# interface TenGigabitEthernet 1/0  
device(conf-if-te-1/0)# switchport  
device(conf-if-te-1/0)# switchport port-security max 10
```

# switchport port-security oui

Configures an Organizationally Unique Identifier (OUI) MAC address for port security on an interface port. All other addresses are ignored

## Syntax

```
switchport port-security oui address
```

```
no switchport port-security oui
```

## Parameters

*address*

The OUI MAC address from which to accept vendor traffic, in the format xxxx.xxxx.xxxx.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no switchport port-security oui** command to disable this option.

The use of static secure MAC addresses is not included in OUI-based port security.

When you configure the first OUI MAC address on a secure port, traffic floods until the entries are programmed in the hardware.



# switchport port-security shutdown-time

Configures the auto recovery time for ports that shuts down following a port security violation on an interface.

## Syntax

```
switchport port-security shutdown-time time
```

## Command Default

Auto recovery of ports is not enabled.

## Parameters

*time*

The amount of time in minutes, the port waits before it recovers from forced port shutdown. Range is from 1 through 15.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The shutdown and no-shutdown processes initiated as part of the port violation action is independent of the shutdown process explicitly initiated by an administrator on the same port on which port MAC security is enabled.

If a port security-based change occurs when a port is shut down, the shutdown timer is not triggered. Consequently, the user must restore the full functionality of the port.

When port security violation causes a port to be shut down and the user manually changes the shutdown time, the shutdown timer is reset and the timer starts with the new shutdown time.

The **no switchport port-security shutdown-time** command disables the auto recovery functionality.

## Examples

The following example configures the auto recovery time as 4 minutes for ports that shuts down following a port security violation on an interface.

```
device(config)# interface TenGigabitEthernet 1/0
device(conf-if-te-1/0)# switchport
device(conf-if-te-1/0)# switchport port-security shutdown-time 4
```

# switchport port-security sticky

Enables sticky MAC learning on the port to convert the dynamically learned MAC addresses to sticky secure MAC addresses.

## Syntax

```
switchport port-security sticky mac-address address vlan vlan_id
```

## Command Default

Sticky MAC learning on the port is not enabled.

## Parameters

**mac-address** *address*

Specifies the MAC address-based VLAN classifier rule used to map to a specific VLAN.

**vlan** *vlan\_id*

Specifies a VLAN.

## Modes

Interface subtype configuration mode

## Usage Guidelines

When sticky MAC learning is enabled on a secured port, the interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All the subsequent sets of dynamically learned MAC addresses will also be converted to sticky secure MAC addresses.

The **no switchport port-security sticky** disables sticky MAC learning on a secure port, and all the sticky MAC addresses will be converted back to dynamically learned MAC addresses.

Sticky MAC addresses persist even if the port goes down or if the device reboots.

## Examples

The following example enables sticky MAC learning on the port and configures port security with sticky MAC address:

```
device(config)# interface TenGigabitEthernet 1/0
device(conf-if-te-1/0)# switchport
device(conf-if-te-1/0)# switchport port-security sticky
switch(conf-if-te-1/0)# switchport port-security sticky mac-address 0000.0018.747C vlan 5
```

# switchport port-security violation

Configures the violation response action for port security on an interface.

## Syntax

```
switchport port-security violation { restrict | shutdown }
```

## Command Default

The port shuts down if port security violation occurs.

## Parameters

### restrict

Drops packets with unknown source addresses until you remove a sufficient number of secure MAC addresses to drop below the maximum value.

### shutdown

Puts the interface into the error-disabled state.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If a MAC address already learned on a secured port ingresses on a non-secured port or through another secured port, it is not considered security violation. In this scenario, MAC movement happens if it is a dynamically learned MAC address. If it is a static MAC address or sticky MAC address, MAC movement does not happen, but the traffic is switched (flooded or forwarded) based on the destination MAC address.

If the port shuts down after security violation, an administrator can explicitly bring up the interface or a shutdown timer can be configured using the **switchport port-security shutdown-time** command. After the configured shutdown time, the interface automatically comes up and the port security configuration remains configured on the port.

When the device reboots after port shutdown due to security violation, the ports come up in the shutdown state.

## Examples

The following example configures the violation response action as shutdown for port security on an interface:

```
device(config)# interface TenGigabitEthernet 1/0
device(conf-if-te-1/0)# switchport
device(conf-if-te-1/0)# switchport port-security violation shutdown
```

# switchport private-vlan association trunk

Assigns a primary private VLAN to private VLAN trunk port.

## Syntax

```
switchport private-vlan association trunk primary_vlan_ID secondary_vlan_ID
```

```
no switchport private-vlan association trunk primary_vlan_ID
```

```
no switchport private-vlan association trunk primary_vlan_ID secondary_vlan_ID
```

## Command Default

The port does not have any PVLAN attributes by default.

## Parameters

*primary\_vlan\_ID*

The primary VLAN identification.

*secondary\_vlan\_ID*

The secondary VLAN identification.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Multiple PVLAN pairs (Primary VLAN, multiple secondaries) can be specified using this command. Therefore, two **no** versions of this command are used to remove association for one primary VLAN, or remove any trunk association.

## Examples

To associate a primary VLAN to PVLAN trunk port, in this example 2 is primary VLAN and 302 is secondary VLAN:

```
switch(conf-if-te-178/0/9)# switchport private-vlan association trunk 2 302
```

To remove a primary VLAN to PVLAN trunk port:

```
switch(conf-if-te-178/0/9)# no switchport private-vlan association trunk 2
```

## Related Commands

[private-vlan](#), [switchport mode private-vlan](#), [switchport private-vlan mapping](#)

# switchport private-vlan host-association

Assigns a secondary and primary VLAN pair to host port.

## Syntax

```
switchport private-vlan host-association primary_vlan_ID secondary_vlan_ID  
no switchport private-vlan host-association
```

## Command Default

The port does not have any PVLAN attributes by default.

## Parameters

*primary\_vlan\_ID*

The primary VLAN identification.

*secondary\_vlan\_ID*

The secondary VLAN identification.

## Modes

Interface subtype configuration mode

## Related Commands

[private-vlan](#), [switchport mode private-vlan](#), [switchport private-vlan mapping](#)

# switchport private-vlan mapping

Maps primary VLAN and secondary VLAN to a promiscuous port.

## Syntax

```
switchport private-vlan mapping primary_vlan_ID [ add | remove ] secondary_vlan  
no switchport private-vlan mapping
```

## Command Default

The port does not have any PVLAN attributes by default.

## Parameters

*primary\_vlan\_ID*

The primary VLAN identification.

**add**

Adds the secondary VLAN to the primary mapping.

**remove**

Removes the secondary VLAN from the primary mapping.

*secondary\_vlan*

The secondary VLAN identification.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command also maps a promiscuous port to selected secondary VLANs. This means only selected VLAN can send packets to this port.

## Related Commands

[private-vlan](#), [switchport private-vlan association trunk](#)

# switchport private-vlan trunk allowed vlan

Adds a VLAN to a private VLAN (PVLAN) trunk port.

## Syntax

```
switchport private-vlan trunk allowed vlan { all | none | [ add | remove | except ] vlan_id } ctag ctag }
no switchport private-vlan trunk allowed vlan vlan_id
```

## Command Default

The port will have default VLAN 1.

## Parameters

<b>all</b>	Allows all VLANs.
<b>none</b>	Removes all VLANs except for VLAN 1.
<b>add</b>	Adds a specified VLAN.
<b>remove</b>	Removes the specified VLAN.
<b>except</b>	Allows all VLANs except the specified VLAN.
<i>vlan_id</i>	Specifies a VLAN.
<b>ctag ctag</b>	Specifies an incoming C-TAG that is associated with a service or transport VF in a Virtual Fabrics context.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **no** form of this command to remove a VLAN or C-TAG from a trunk port.

For service or transport VFs (VLAN ID 4096 through 8191), the C-TAG cannot be a default VLAN, a reserved VLAN, and FCoE VLAN, or an internal control VLAN. Examples

The following illustrates the configuration of PVLANS for both 802.1Q VLANs and service or transport VFs in a Virtual Fabrics context.

switchport private-vlan trunk allowed vlan

Configure a PVLAN trunk port:

```
switch(config)# int te 4/1
switch(config-if-te-2/0/2)# switchport mode private-vlan trunk
```

Configure 802.1Q VLANs and service or transport VFs in a Virtual Fabrics context:

```
switch(config-if-te-2/0/2)# switchport private-vlan trunk allowed vlan add 400
switch(config-if-te-2/0/2)# switchport private-vlan trunk allowed vlan add 5000 ctag 100
```

Configure service or transport VFs as PVLANs, by using the **switchport private-vlan association** command:

```
switch(config-if-te-2/0/2)# switchport private-vlan association trunk 6000 7000
switch(config-if-te-2/0/2)# switchport private-vlan association trunk 6000 8000
```

## Related Commands

[switchport mode](#), [switchport private-vlan association trunk](#)



# switchport private-vlan trunk native-vlan

Sets native private VLAN (PVLAN) characteristics of the Layer 2 trunk interface for classifying untagged traffic.

## Syntax

```
switchport private-vlan trunk native-vlan vlan_id  
no switchport private-vlan trunk native-vlan
```

## Parameters

*vlan\_id*  
Specifies a VLAN to transmit and receive through the Layer 2 interface.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no switchport trunk native-vlan** to reset the native VLAN to the default setting.

Native VLAN configuration is not supported for a port in private vlan trunk promiscuous mode.

## Examples

To set native PVLAN characteristics for a VLAN whose VLAN ID is 120:

```
switch(config)# interface tengigabitethernet 178/0/9  
switch(conf-if-te-178/0/9)# switchport private-vlan trunk native-vlan 120
```

## Related Commands

[switchport mode private-vlan](#), [switchport private-vlan association trunk](#), [switchport private-vlan trunk allowed vlan](#)

# switchport trunk allowed vlan rspan-vlan

Adds or removes VLANs on a Layer 2 interface in trunk mode.

## Syntax

```
switchport trunk allowed { vlan | rspan-vlan } { add vlan_id { ctag { id | ctag - range } | all | except vlan_id | none | remove
vlan_id }
```

## Parameters

### add *vlan\_id*

Adds a VLAN to transmit and receive through the Layer 2 interface. The VLAN can be an 802.1Q VLAN, an RSPAN VLAN, or a transport VLAN.

### all

Allows only 802.1Q VLANs to transmit and receive through the Layer 2 interface. This keyword does not apply to classified or transport VLANs.

### ctag

Specifies an incoming C-TAG or range of C-TAGs for classified or transport VLANs in a Virtual Fabrics context.

*id*

C-TAG ID.

*range*

Range of C-TAG IDs, for example, 100-200, or 10,20,100-200, applicable only if the VLAN is a transport VLAN.

### except *vlan\_id*

Allows only 802.1Q VLANs except the specified VLAN ID to transmit and receive through the Layer 2 interface.

### none

Allows only 802.1Q VLANs to transmit and receive through the Layer 2 interface. This keyword does not apply to service or transport VFs in a Virtual Fabrics context.

### rspan-vlan *vlan\_id*

Selects a VLAN for Remote Switched Port Analyzer (RSPAN) traffic monitoring.

### remove *vlan\_id*

Removes a VLAN that transmits and receives through the Layer 2 interface.

## Modes

Interface subtype configuration mode

## Usage Guidelines

For service or transport VFs (VLAN ID 4096 through 8191), the C-TAG cannot be a default VLAN, a reserved VLAN, and FCoE VLAN, or an internal control VLAN.

A transport VF C-TAG can be any VLAN ID that is not used in other classifications or as a 802.1Q VLAN.

## Examples

To add the tagged VLAN 100 to a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(config-if-te-178/0/9)# switchport trunk allowed vlan add 100
```

To remove the tagged VLAN 100 from the interface:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(config-if-te-178/0/9)# switchport trunk allowed vlan remove 100
```

The following examples illustrate configuration in a Virtual Fabrics context:

Configure an interface as a trunk switchport.

```
switch(config)# int te 1/0/1
switch(config-if-te-1/0/1)# switchport mode trunk
```

A C-TAG is required for a classified VLAN (VLAN ID from 4096 through 8191):

```
switch(config-if-te-1/0/1)# switchport trunk allowed vlan add 7000
switch(config-if-te-1/0/1)# syntax error: unknown argument
```

Configure a classified VLAN with a C-TAG:

```
switch(config-if-te-1/0/1)# switchport trunk allowed vlan add 5000 ctag 100
switch(config-if-te-1/0/1)# switchport trunk allowed vlan add 6000 ctag 200
```

An 802.1Q vlan specified as a user VLAN cannot be used as a C-TAG in a classified VLAN. The following show conflicts.

- Edge C-TAG 100 is already assigned to VLAN 5000 at the same port:

```
switch(config-if-te-1/0/1)# switchport trunk allow vlan add 8000 ctag 100
switch(config-if-te-1/0/1)# %Error: C-tag is already used.
```

- Edge VLAN 100 is already used as a C-TAG in a classified VLAN:

```
switch(config-if-te-1/0/1)# switchport trunk allow vlan 100
switch(config-if-te-1/0/1)# %%Error: One of the vlans in the range is configured as a ctag on the same
port.
switch(config-if-te-1/0/1)# switchport trunk allow vlan all
switch(config-if-te-1/0/1)# %%Error: Virtual-fabric vlan classification configuration exists.
switch(config-if-te-1/0/1)# switchport trunk allow vlan add 888
```

- Edge VLAN 888 was already used in 802.1Q configuration.

```
switch(config-if-te-1/0/1)# switchport trunk allow vlan add 8000 ctag 888
switch(config-if-te-1/0/1)# %Error: Ctag is configured in the allowed range on this port.
```

# switchport trunk default-vlan

Configures tagged or untagged data traffic that does not match any classification rule on a trunk port, supporting service or transport VFs in a Virtual Fabrics context.

## Syntax

```
switchport trunk default-vlan vlan_id  
no switchport trunk default-vlan vlan_id
```

## Parameters

*vlan\_id*

Adds a classified VLAN (VLAN ID > 4095) to transmit and receive through the Layer 2 interface.

## Modes

Interface subtype configuration mode on a trunk port

## Usage Guidelines

Enter **no switchport trunk default-vlan *vlan\_id*** to remove the default VLAN configuration.

## Examples

Create a transport VF in a Virtual Fabrics context:

```
switch(config)# interface vlan 6000  
switch(config-vlan-6000)# transport-service 60
```

Classify all nonmatching traffic except native VLAN traffic to the transparent default VLAN:

```
switch(config-if-te-2/0/1)# switchport trunk default-vlan 6000
```

# switchport trunk native-vlan

Sets native VLAN characteristics as an 802.1Q VLAN, or, in a Virtual Fabrics context, as service or transport VF on a trunk port, matching tagged or untagged data traffic that does not match a classification rule.

## Syntax

```
switchport trunk native-vlan vlan_id [ ctag id ]
no switchport trunk native-vlan vlan_id [ ctag id ]
```

## Parameters

*vlan\_id*

Adds a VLAN to transmit and receive through the Layer 2 interface.

*ctag id*

Sets an optional C-TAG for a service or transport VF (VLAN ID > 4095). If not present, the native VLAN is untagged.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Note the following:

- For VLAN IDs above 4095, the **ctag** keyword is optional.
- If **ctag** is not used, the native VLAN is untagged and the command is validated against the **[no] switchport trunk tag native-vlan** command, which controls the tagging of the native VLAN at the interface level. The **switchport trunk native-vlan** command is accepted only if the configuration set by the **switchport trunk tag native-vlan** command allows untagged packets. For VLAN IDs above 4095, validation against the global command **no vlan dot1q tag native** is not required.
- The native VLAN must accept tagged frames for the **ctag** keyword to apply.
- For 802.1Q VLANs (VLAN ID < 4096), both the interface subtype and global commands that control native VLAN tagging apply to the specified native VLAN.

Use the **no** form of this command to unconfigure the native VLAN. VLAN 1 then becomes the native VLAN.

For service or transport VFs (VLAN ID 4096 through 8191), the C-TAG cannot be a default VLAN, a reserved VLAN, or an internal control VLAN. An FCoE VLAN ID can be used as a C-TAG provided the interface is not configured for "fcoeport default."

Enter **no switchport trunk native-vlan** to reset the native VLAN to the default setting

## Examples

To set native VLAN characteristics for an 802.1Q VLAN whose VLAN ID is 120:

```
switch(config)# interface tengigabitethernet 178/0/9
switch(conf-if-te-178/0/9)# switchport trunk native-vlan 120
```

The following illustrates the use of the command in a Virtual Fabrics context:

- Configure an interface as a switchport trunk and set the tagging of the native VLAN at the interface level:

```
switch(config)# int te 2/0/1
switch(config-if-te-2/0/1)# switchport mode trunk
switch(config-if-te-2/0/1)# switchport trunk tag native-vlan
```

- Change the native VLAN from the default of 1 to a classified VLAN (VLAN ID > 4095) and add an optional C-TAG:

```
switch(config-if-te-2/0/1)# switchport trunk native-vlan 5000 ctags 50
```

- Change the new native default VLAN to an 802.1Q VLAN (VLAN ID < 4096):

```
switch(config-if-te-2/0/1)# switchport trunk native-vlan 200
```

- The interface must allow untagged packets for classified native VLANs without a C-TAG:

```
switch(config-if-te-2/0/1)# switchport trunk native-vlan 5000
%%Error: Cannot configure non-dot1q native-vlan without a ctags, when native-vlan-tagging is enabled.
switch(config-if-te-2/0/1)# no
switchport trunk tag native-vlan
switch(config-if-te-2/0/1)# switchport trunk native-vlan 5000
```

# switchport trunk native-vlan-untagged

Configures a port to accept only untagged packets, and specifies that those packets be egress untagged in a Virtual Fabrics context. The untagged packets may be classified to an 802.1Q VLAN, a service VF, or a transport VF.

## Syntax

```
switchport trunk native-vlan-untagged vlan_id
```

```
no switchport trunk native-vlan-untagged
```

## Parameters

*vlan\_id*

Adds a classified VLAN (VLAN ID > 4095) to transmit and receive through the Layer 2 interface.

## Modes

Interface subtype configuration mode on a trunk port

## Usage Guidelines

This command is supported when the port is in no-default-vlan trunk mode, as enabled by means of the **switchport mode trunk-no-default-native** command.

Use the **no switchport trunk native-vlan-untagged** command to remove the configuration.

Port mode change is not allowed when port security is enabled on the interface.

## Examples

Configure untagged native VLAN 5000, allow VLAN 6000, and make VLAN 7000 the default VLAN.

```
switch(config)# interface te 2/1/1
switch(config-if-te-2/1/1)# switchport mode trunk-no-default-native
switch(config-if-te-2/1/1)# switchport trunk native-vlan untagged 5000
switch(config-if-te-2/1/1)# switchport trunk add vlan 6000 ctag 100-200
switch(config-if-te-2/1/1)# switchport trunk default-vlan 7000
```

Remove the native VLAN 5000.

```
switch(config-if-te-2/1/1)# no switchport trunk native-vlan-untagged
```

## Related Commands

[switchport mode trunk-no-default-native](#), [switchport trunk native-vlan-xtagged](#)

# switchport trunk native-vlan-xtagged

Configures a port to accept both tagged and untagged packets, and specifies the egress tagging behavior in a Virtual Fabrics context.

## Syntax

```
switchport trunk native-vlan-xtagged vlan_id [ ctag cvid ] egress { tagged | untagged | any }
```

```
no switchport trunk native-vlan-xtagged
```

## Parameters

*vlan\_id*

Adds a classified VLAN (VLAN ID > 4095) to transmit and receive through the Layer 2 interface.

**ctag** *cvid*

Sets an optional C-TAG (802.1Q VLAN ID) for a service or transport VF (VLAN ID > 4095).

**egress**

Enables the selection of required tagging options.

**tagged**

Specifies packets as tagged.

**untagged**

Specifies packets as untagged.

**any**

Specifies that packets preserve their ingress encapsulation.

## Modes

Interface subtype configuration mode on a trunk port

## Usage Guidelines

This command is supported when the port is in no-default-vlan trunk mode, as enabled by means of the **switchport mode trunk-no-default-native** command.

Note the following:

- Ingress packets may be classified to an 802.1Q VLAN, a service VF, or a transport VF.
- The native VLAN must accept tagged frames for the **ctag** keyword to apply.
- If the specified VLAN is an 802.1Q VLAN, the **ctag** option is not required.
- If the specified VLAN is an 802.1Q VLAN or a service VF, the **egress** tagging options are **tagged** or **untagged**.
- If the specified VLAN is a transport VF, then the **egress** tagging option must be **any** to preserve the encapsulation of ingress frames.

Use the **no switchport trunk native-vlan-xtagged** command to remove the configuration.

Port mode change is not allowed when port security is enabled on the interface.



## Examples

Configure transport VF 6000 that accepts C-TAG range 100 through 200 and a native VLAN that can be either tagged or untagged.

```
switch(config)# interface te 2/1/1
switch(config-if-te-2/1/1)# switchport mode trunk-no-default-native
switch(config-if-te-2/1/1)# switchport trunk native-vlan-xtagged 6000 ctag 10 egress any
switch(config-if-te-2/1/1)# switchport trunk allow vlan 6000 ctag 100-200
```

Remove the native VLAN from the transport VF.

```
switch(config-if-te-2/1/1)# no switchport trunk native-vlan-xtagged
```

## Related Commands

[switchport mode trunk-no-default-native](#), [switchport trunk native-vlan-untagged](#)

# switchport trunk tag native-vlan

Enables tagging on native VLAN traffic.

## Syntax

```
switchport trunk tag native-vlan
```

```
no switchport trunk tag native
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no switchport trunk tag native** to untag native traffic for a specific interface.

## Examples

To enable tagging for native traffic on a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/9
```

```
switch(conf-if-te-178/0/9)# switchport trunk tag native-vlan
```

# system tunnel suppress-debounce

Suppresses the debounce-timer functionality for a tunnel.

## Syntax

```
system tunnel suppress-debounce
no system tunnel suppress-debounce
```

## Command Default

There is a one second delay after the underlay network stops before the tunnel stops.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no system tunnel suppress-debounce** command to remove the suppression of the delay.

The debounce timer is a one second delay after the underlay network stops before the tunnel stops. This command suppresses that one second delay, so the underlay network and tunnel stop simultaneously.

## Examples

Typical command execution.

```
device(config)# system tunnel suppress-debounce
device(config)#
```

## History

Release version	Command history
5.0.2b	This command was introduced.

# system-description

Sets the global system description specific to LLDP.

## Syntax

`system-description` *line*

`no system-description`

## Parameters

*line*

Specifies a description for the LLDP system. The string must be between 1 and 50 ASCII characters in length.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

Enter `no system-description` to clear the global LLDP system description.

## Examples

To set the global system description specific to LLDP:

```
device(conf-lldp)# system-description NetworkOS
```

## Related Commands

[system-name](#)

# system-id oui

Configures the system ID OUI to start with either 00.e0.52 or 01.e0.52.

## Syntax

```
system-id oui { 01.e0.52 | 00.e0.52 }  
no system-id oui
```

## Command Default

The default value is 01.e0.52.

## Parameters

**01.e0.52**  
This is the default system ID OUI.

**00.e0.52**  
This is the alternate system ID OUI.

## Modes

Spanning tree configuration mode

## Usage Guidelines

This command sets the first three bytes of the bridge system ID to either 01.e0.52 or 00.e0.52. The bridge system ID is formatted as "01.e0.52.<vcs-id>.<random-number>". The bridge ID is only generated when STP is in the "no shutdown" state. The bridge ID transmitted to MSTPD.

This command applies to all STP modes.

The **no system-id oui** command sets the system ID to the default value of 01.e0.52. If spanning-tree is in the "no shutdown" state then the new system ID is configured and STP re-converges.

When this configuration is present, downgrading to a previous version is blocked.

## Examples

Typical command example with help output.

```
device# configure terminal  
device(config)# protocol spanning-tree mstp  
device(config-mstp)# system-id oui 00.e0.52
```

## History

Release version	Command history
7.0.0b	This command was introduced.

# system-mode maintenance

Enables maintenance mode for graceful traffic diversion from the switch by shutting down edge ports and increasing the cost of ISLs, to facilitate debugging or firmware upgrades.

## Syntax

```
system-mode maintenance
no system-mode maintenance
```

## Command Default

Maintenance mode is disabled.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This command is persistent across reboots. This command and the **chassis disable** command are mutually exclusive.

Use the **no** form of this command to disable maintenance mode and revert to the previous user-configured administrative state of the ports.

## Examples

To enable maintenance mode:

```
device# config terminal
device(cocnfig)# rbridge-id 1
device(config-rbridge-id-1)# system-mode maintenance
```

To disable maintenance mode:

```
device# config terminal
device(cocnfig)# rbridge-id 1
device(config-rbridge-id-1)# no system-mode maintenance
```

## History

Release version	Command history
7.0.0	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

# system-monitor

Manages the monitoring of FRUs and sets a variety of alerts when thresholds are exceeded.

## Syntax

```
system-monitor { LineCard [ alert [ action [ all | email | none | raslog ] ] | state [ all | faulty | inserted | none | on | removed ] ] |
  threshold [ down-threshold | marginal-threshold ] ] | MM [ threshold [ down-threshold | marginal-threshold ] ] | cid-card
  [ alert [ action | state [ all | faulty | inserted | none | on | removed ] ] | threshold [ down-threshold | marginal-threshold ] ] |
  compact-flash [ threshold [ down-threshold | marginal-threshold ] ] | fan [ alert [ action | state [ all | faulty | inserted |
  none | on | removed ] ] | threshold [ down-threshold | marginal-threshold ] ] | power [ alert [ action | state [ all | faulty |
  inserted | none | on | removed ] ] | threshold [ down-threshold | marginal-threshold ] ] | sfp [ alert [ action state ] ] temp
  [ threshold [ down-threshold | marginal-threshold ] ] }
```

```
no system-monitor
```

## Command Default

For system monitoring defaults, see the "System Monitor" chapter in the *Network OS Administrator's Guide Supporting Network OS v.4.0.0*.

## Parameters

### LineCard

Specifies alerts and thresholds for line cards.

### MM

Specifies thresholds for management modules.

### cid-card

Specifies alerts and thresholds for the chassis ID card.

### compact-flash

Specifies thresholds for the compact flash device.

### fan

Specifies alerts and thresholds for the fans.

### power

Specifies alerts and thresholds for the power supplies.

### sfp

Specifies alerts for the small form-factor pluggable devices.

### temp

Specifies thresholds for the temperature sensors.

### alert

Specifies whether an alert is sent when a threshold value is either above or below a threshold trigger.

### action

Specifies the response type.



**all**  
Specifies that e-mail and RASLog messaging are used.

**email**  
Specifies that an e-mail message is sent.

**none**  
Specifies that no message is sent.

**raslog**  
Specifies RASLog messaging.

**state**  
Specifies the hardware state to be monitored.

**all**  
Specifies that all hardware states are monitored.

**faulty**  
Specifies that hardware is monitored for faults.

**inserted**  
Specifies that the insertion state of hardware is monitored.

**none**  
Specifies that no hardware states are monitored.

**on**  
Specifies that the hardware on/off state is monitored.

**removed**  
Specifies that the removal of hardware is monitored.

**threshold**  
Specifies the monitoring of thresholds

**down-threshold**  
Specifies an integer value that, when exceeded, indicates when hardware is down.

**marginal-threshold**  
Specifies an integer value that, when exceeded, indicates when hardware is operating marginally.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Use this command to configure field-replaceable unit (FRU) monitoring and actions. Depending on these configuration settings, a variety of actions are generated when there is a change in FRU state.

Use this command in RBridge subconfiguration mode to manage the system health monitoring of individual nodes in a cluster.

## Examples

```
switch(config-rbridge-id-154)# system-monitor sfm threshold down-threshold 3 marginal-threshold 2  
switch(config-rbridge-id-154)# system-monitor cid-card alert state faultyinserted action email
```

## Related Commands

[rbridge-id](#), [show system monitor](#), [system-monitor-mail](#)

# system-monitor-mail

Configures Fabric Watch e-mail alerts on the device.

## Syntax

```
system-monitor-mail { fru | interface | relay { host_ip | domain_name } | security | sfp } enable | email-id ]
no system-monitor-mail
```

## Command Default

The default source is disabled.

## Parameters

### fru

Configures e-mail alerts for FRUs.

### interface

Configures e-mail alerts for interfaces.

### relay

Configures the relay host for e-mail to work in a non-DNS environment.

#### *host\_ip*

Specifies the IPv4 address of the mail server.

#### *domain\_name*

Specifies the domain that corresponds to the e-mail ID.

### security

Configures e-mail alerts for security.

### sfp

Configures e-mail alerts for SFPs.

### enable

Enables or disables e-mail alerts for the above options.

#### *email-id*

Specifies the e-mail address to where the alert will be sent.

## Modes

Global configuration mode

## Usage Guidelines

For an e-mail alert to function correctly, add the IP addresses and host names to DNS in addition to configuring the domain name and name servers. Both relay parameters (the host IP address and the domain name) must be configured in a non-DNS environment. In a DNS environment, only the host IP address is required).

## Examples

```
device(config)# system-monitor-mail ?
```

Possible completions:

```
fru          Configure FRU mail settings
interface    Configure interface mail settings
relay        Configure relay ip mail settings
security     Configure security mail settings
sfp          Configure sfp mail settings
device(config)# system-monitor-mail fru enable
```

```
device(config)# system-monitor-mail relay ?
```

Possible completions:

```
<host-ip:IP address> <host-ip:string, min: 1 chars, max: 253 chars>
device(config)# system-monitor-mail relay 1.2.3.4 ?
```

Possible completions:

```
domain-name  Domain name server
device(config)# system-monitor-mail relay 1.2.3.4 domain-name ?
```

Possible completions:

```
<LINE:0-64> Domain name[]
device(config)# system-monitor-mail relay 1.2.3.4 domain-name abc.extremenetworks.com
```

```
device# show running-config system-monitor-mail relay
```

```
system-monitor-mail relay 1.2.3.4 domain-name abc.extremenetworks.com
```

To create a mapping:

```
device(config)# system-monitor-mail relay host-ip 1.2.3.4 domain-name abc.extremenetworks.com
```

To delete the mapping:

```
device(config)# no system-monitor-mail relay host-ip 1.2.3.4
```

To change the domain name:

```
device(config)# system-monitor-mail relay host-ip 1.2.3.4 domain-name mail.extremenetworks.com
```

## Related Commands

[show system monitor](#), [system-monitor](#)

# system-name

Sets the global system name specific to LLDP.

## Syntax

**system-name** *name*

**no system-name**

## Command Default

The host name from the device is used.

## Parameters

*name*

Specifies a system name for the LLDP. The string must be between 1 and 32 ASCII characters in length.

## Modes

Protocol LLDP configuration mode

## Usage Guidelines

Enter **no system-name** to delete the name.

## Examples

To specify a system name for the LLDP:

```
device(conf-lldp)# system-name System10
```

## Related Commands

[system-description](#)

# table-map

Maps external entry attributes into the BGP routing table, ensuring that those attributes are preserved after being redistributed into OSPF.

## Syntax

**table-map** *string*

**no table-map** *string*

## Command Default

This option is disabled.

## Parameters

*string*

Specifies a route map to be whose attributes are to be preserved. Range is from 1 through 63 ASCII characters.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to remove the table map.

Use this command only to set the tag values. Normally, a route map is applied on routes (and therefore the routes are updated) before it is stored in the BGP routing table. Use the **table-map** command to begin the update before the routes are stored in the IP routing table.

Configurations made by this command apply to all peers.

Route maps that contain **set** statements change values in routes when the routes are accepted by the route map. For inbound route maps (route maps that filter routes received from neighbors), the routes are changed before they enter the BGP4 routing table. For tag values, if you do not want the value to change until a route enters the IP routing table, you can use a table map to change the value. A table map is a route map that you have associated with the IP routing table. The device applies the **set** statements for tag values in the table map to routes before adding them to the routing table. To configure a table map, you first configure the route map, then identify it as a table map. The table map does not require separate configuration. You can have only one table map.

**NOTE**

Use table maps only for setting the tag value. Do not use table maps to set other attributes. To set other route attributes, use route maps or filters. To create a route map and identify it as a table map, enter commands such those shown in the first example below. These commands create a route map that uses an address filter. For routes that match the IP prefix list filter, the route map changes the tag value to 100 and is then considered as a table map. This route map is applied only to routes that the device places in the IP routing table. The route map is not applied to all routes. The first example below assumes that IP prefix list p11 has already been configured.

## Examples

This example illustrates the execution of the **table-map** command.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# route-map tag_ip permit 1
device(config-route-map/tag_ip/permit/1)# match ip address prefix-list p11
device(config-route-map/tag_ip/permit/1)# set tag 100
device(config-route-map/tag_ip/permit/1)# exit
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# table-map tag_ip
```

This example removes the table map for the default VRF.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# no table-map tag_ip
```

This example removes the table map for VRF "red".

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# no table-map tag_ip
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast VRF configuration modes.

## Related Commands

[route-map](#)

# tacacs-server

Configures a Terminal Access Controller Access-Control System plus (TACACS+) server.

## Syntax

```
tacacs-server { host hostname [ use-vrf vrf-name ]
tacacs-server { source-ip [ chassis-ip | mm-ip ] }
[ port portnum ]
[ protocol { chap | pap } ]
[ key shared_secret ]
[ encryption-level value_level ]
[ timeout secs ]
[ retries num ]
no tacacs-server { host hostname | source-ip [ chassis-ip | mm-ip ] } [ use-vrf vrf-name ]
```

## Command Default

Refer to the Parameters section for specific defaults.

## Parameters

- host** *hostname*  
Specifies the IP address or domain name of the TACACS+ server. IPv4 and IPv6 addresses are supported.
- use-vrf** *vrf-name*  
Specifies a VRF through which to communicate with the TACACS+ server. See the Usage Guidelines.
- tacacs-server source-ip** [*chassis-ip* | *mm-ip* ]  
Specifies the chassis IP address or MM IP address as the source IP address for TACACS+ authentication and accounting.
- port** *portnum*  
Specifies the authentication port. Valid values range from 0 through 65535. The default is 49.
- protocol** { *chap* | *pap* }  
Specifies the authentication protocol. Options include CHAP and PAP. The default is CHAP.
- key** *shared\_secret*  
Specifies the text string that is used as the shared secret between the device and the TACACS+ server to make the message exchange secure. The key must be between 1 and 40 characters in length. The default key is **sharedsecret** . The exclamation mark (!) is supported both in RADIUS and TACACS+ servers, and you can specify the password in either double quotes or the escape character (\), for example "**secret!key**" or **secret\!key**. The only other valid characters are alphanumeric characters (such as a-z and 0-9) and underscores. No other special characters are allowed.



**encryption-level** *value\_level*

Designates the encryption level for the shared secret key operation. This operand supports JITC certification and compliance. The valid values are 0 and 7, with 0 being clear text and 7 being the most heavily encrypted. The default value is 7.

**timeout** *secs*

Specifies the time to wait for the TACACS+ server to respond. The default is 5 seconds.

**retries** *num*

Specifies the number of attempts allowed to connect to a TACACS+ server. The default is 5 attempts.

## Modes

Global configuration mode

## Usage Guidelines

If a TACACS+ server with the specified IP address or host name does not exist, it is added to the server list. If the TACACS+ server already exists, this command modifies the configuration. The **key** parameter does not support an empty string.

Executing the **no** form of the **tacacs-server** command attributes resets the specified attributes to their default values.

### NOTE

Before downgrading to a software version that does not support the **encryption-level** keyword, set the value of this keyword to **0**. Otherwise, the firmware download will throw an error that requests this value be set to **0**.

Before downgrading to a version that doesn't support **tacacs-server source-ip**, you must remove the source-ip configuration using **no tacacs-server source-ip**. Otherwise, the firmware download process throws an error requesting to reset the cipher.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

To configure an IPv4 TACACS+ server:

```
device# configure terminal
device(config)# tacacs-server host 10.24.65.6
device(config-host-10.24.65.6/mgmt-vrf)# tacacs-server source-ip chassis-ip
device(config-host-10.24.65.6/mgmt-vrf)# protocol chap retries 100
device(config-host-10.24.65.6/mgmt-vrf)#
```

To modify an existing TACACS+ server configuration:

```
device# configure terminal
device(config)# tacacs-server host 10.24.65.6
device(config-tacacs-server-10.24.65.6/mgmt-vrf)# key "changedsec"
```

To delete a TACACS+ server:

```
device# configure terminal
device(config)# no tacacs-server host 10.24.65.6
```

To configure an IPv6 TACACS+ server:

```
device# configure terminal
device(config)# tacacs-server host fec0:60:69bc:94:211:25ff:fec4:6010
device(config-tacacs-server-fec0:60:69bc:94:211:25ff:fec4:6010/mgmt-vrf)# protocol chap key "mysecret"
device(config-tacacs-server-fec0:60:69bc:94:211:25ff:fec4:6010/mgmt-vrf)# tacacs-server source-ip
chassis-ip
device(config-tacacs-server-fec0:60:69bc:94:211:25ff:fec4:6010/mgmt-vrf)#
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.

## Related Commands

[radius-server](#), [show running-config radius-server](#), [show running-config tacacs-server](#)

# tcp burstrate

Sets the threshold for the burst rate of TCP traffic, and defines the lockout time once that threshold is passed.

## Syntax

```
tcp burstrate packet lockup seconds
```

```
no tcp burstrate
```

## Command Default

This feature is disabled.

## Parameters

*packet*

The maximum number of packets allowed over five seconds. Range is from 1 through 100000.

**lockup** *seconds*

Sets the number of seconds to lock up the port. Range is from 1 through 3000.

## Modes

Global configuration mode

## Usage Guidelines

To protect against TCP SYN attacks, you can configure the Extreme device to drop TCP SYN packets when excessive numbers are encountered. You can set threshold values for TCP SYN packets that are targeted at the router itself or passing through an interface, and drop them when the thresholds are exceeded.

This command sets the threshold for the burstrate, and defines the lockout time once that threshold is passed.

# telnet

Establishes a Telnet session to a remote networking device.

## Syntax

```
telnet { IP_address | hostname }
telnet interface port-num { interface | vrf vrf-name } { IP_address | hostname }
telnet interface <N>gigabitethernet { IP_address | hostname | vrf vrf-name }
telnet interface management { IP_address | hostname | vrf vrf-name }
telnet interface ve vlan-id hostname
telnet interface vrf vrf-name
telnet vrf vrf-name
```

## Command Default

The default port is 23.

## Parameters

*IP\_address*

The server IP address in either IPv4 or IPv6 format.

*hostname*

The host name (a string between 1 and 63 ASCII characters in length).

**interface**

Specifies an interface.

<N> **gigabitethernet**

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>**gigabitethernet** with the desired operand (for example, **tengigabitethernet** specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

**management**

Specifies a management interface.

**port-number** *port*

Specifies the port number in the remote device to connect to. Range is from 0 through 65535. For the connection to succeed, a TCP server must be listening for client connections at the specified port.

**ve** *vlan-id*

Range is from 1 through 4090 if Virtual Fabrics is disabled, and from 1 through 8191 if Virtual Fabrics is enabled.

**vrf** *vrf-name*

Specifies a VRF instance. See the Usage Guidelines.

## Modes

Privileged EXEC mode

RBRidge ID configuration mode

## Usage Guidelines

To use the **telnet** command on the management VRF, use the **vrf** keyword and enter "mgmt-vrf" manually.

The following features are not supported:

- Display Telnet sessions
- Ability to terminate hung Telnet sessions

## Examples

To establish a Telnet session from a switch to a remote networking device:

```
device# telnet 10.17.37.157

Trying 10.17.37.157...
Connected to 10.17.37.157.
Escape character is '^]'.
Network OS (sw0)
sw0 login:
```

To establish a Telnet session from a switch to a remote networking device with port number 22 and the management VRF:

```
device# telnet 10.17.37.157 port-number 22 vrf mgmt-vrf
```

In a VCS Fabric, this command also functions in RBRidge ID mode.

```
switch# rbridge-id 3
switch(config-rbridge-id-3)# telnet server shutdown
```

# telnet server shutdown

Disables Telnet service on the switch.

## Syntax

```
telnet server [ use-vrf vrf-name ] shutdown
no telnet server [ use-vrf vrf-name ] shutdown
```

## Parameters

**use-vrf** *vrf-name*  
Specifies a user-defined VRF.

## Modes

RBridge ID configuration mode

## Usage Guidelines

Enter **no telnet server shutdown** to enable Telnet service. This command is not distributed across a cluster. The RBridge ID of the node should be used to configure service on individual nodes.

The use of the **use-vrf** keyword brings down the server only for the specified VRF. The user can shut down any server in any VRF, including the management and default VRF.

When this command is executed and a VRF is not specified by means of the **use-vrf** keyword, the server is brought down only in the management VRF ("mgmt-vrf") (the default VRF for this command).

When this command is executed at the RBridge ID level for a specified VRF, connectivity to servers in that VRF is enabled, whereas this service for nonspecified R Bridges is shut down.

## Examples

To shut down Telnet service on an RBridge on the management VRF:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# telnet server shutdown
```

To shut down Telnet service on an RBridge for a user-defined VRF:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# telnet server use-vrf myvrf shutdown
```

To enable Telnet service on an RBridge on the management VRF:

```
device(config)# rbridge-id 3
device(config-rbridge-id-3)# no telnet server shutdown
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.

## Related Commands

[show running-config telnet server](#), [show telnet server status](#)

# telnet server standby enable

Enables the Telnet services on the standby MM.

## Syntax

**telnet server standby enable**

**no telnet server standby enable**

## Command Default

The Telnet services are disabled on the standby MM.

## Modes

RBridge ID configuration mode

## Usage Guidelines

The **no telnet server standby enable** command disables the Telnet services on the standby MM.

## Examples

Typical command output:

```
switch(config-rbridge-id-1)# do show running-config rbridge-id | include standby
% No entries found.
switch(config-rbridge-id-1)# telnet server standby enable
switch(config-rbridge-id-1)# do show running-config rbridge-id | include standby
telnet server standby enable
switch(config-rbridge-id-1)#
```

Typical command output:

```
switch(config-rbridge-id-1)# no telnet server standby enable
switch(config-rbridge-id-1)# do show running-config rbridge-id | include standby
% No entries found.
switch(config-rbridge-id-1)#
```

## History

Release version	Command history
5.0.1a	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[ssh server standby enable](#)



# terminal

Sets terminal parameters for the current session.

## Syntax

```
terminal [ length number_of_lines ] [ monitor ] [ timeout value ]  
no terminal [ length ] [ monitor ] [ timeout ]
```

## Command Default

The default for **length** is 24.

## Parameters

**length** *number\_of\_lines*

Specifies the number of lines to be displayed. Valid values range from 1 through 512. Specify 0 for infinite length.

**monitor**

Enables terminal monitoring.

**timeout** *value*

Specifies the timeout value in minutes. Valid values range from 0 through 136. Specify 0 to disable timeout.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported only on the local switch.

This command overrides the timeout configuration set by the **line vty exec-timeout** command, but only for the duration of the current session. When the current session ends, the configured values apply for any subsequent sessions.

This command is not available on the standby management module.

Enter **no terminal** (optionally with a specific parameter) to restore the current terminal settings to default.

## Examples

To set the display length to 30 lines:

```
switch# terminal length 30
```

To set the timeout length to 60 minutes:

```
switch# terminal timeout 60
```

To restore all settings to default values:

```
switch# no terminal
```

To restore only the timeout setting to its default values:

```
switch# no terminal timeout
```

## Related Commands

[line vty exec-timeout](#)

# threshold-monitor cpu

Configures monitoring of CPU usage of the system and alerts the user when configured thresholds are exceeded.

## Syntax

```
threshold-monitor cpu [ [ actions [ none | raslog [ { limit limit_when_reached | poll polling_interval | retry
  number_of_retries ] ] ] ] }
```

```
no threshold-monitor cpu
```

## Parameters

### actions

Specifies the action to be taken when a threshold is exceeded.

### none

No action is taken.

### raslog

Specifies RASLog messaging.

### limit

Specifies the baseline CPU usage limit as a percentage of available resources.

#### *limit\_when\_reached*

When the limit set by this parameter is exceeded, a RASLog WARNING message is sent. When the usage returns below the limit, a RASLog INFO message is sent. Valid values range from 0 through 80 percent. The default is 70 percent.

### poll

Specifies the polling interval in seconds.

#### *polling\_interval*

The range is from 0 through 3600. The default is 120

### retry

Specifies the number of polling retries before desired action is taken.

#### *number\_of\_retries*

Range is from 1 through 100. The default is 3.

## Modes

RBridge ID configuration mode

## Usage Guidelines

This command sends a RASLog WARNING message when configured thresholds are exceeded.

## Examples

```
device(config-rbridge-id-154)# threshold-monitor cpu actions rasloglimit 50 poll10
```

## Related Commands

[rbridge-id](#)

# threshold-monitor interface

Configures monitoring of port statistics on all external gigabit Ethernet interfaces: 1 GbE, 10 GbE, and 40 GbE.

## Syntax

```
threshold-monitor interface { [ apply policy_name | pause | policy policy_name ] type Ethernet area [ CRCAlignErrors [ alert
[ above [ highthresh-action [ [ all | lowthresh-action ] | email | fence | none | raslog ] | lowthresh-action [ all | email none |
raslog ] | below [ highthresh-action [ all | email | fence | none raslog ] | lowthresh-action [ all | email | none | raslog ] ] |
threshold [ buffer | high-threshold | low-threshold | timebase [ day | hour | minute | none ] ] | IFG [ alert [ above
[ highthresh-action [ [ all | lowthresh-action ] | email | fence | none | raslog ] | lowthresh-action [ all | email none | raslog ] ] |
below [ highthresh-action [ all | email | fence | none raslog ] | lowthresh-action [ all | email | none | raslog ] ] | threshold
[ buffer | high-threshold | low-threshold | timebase [ day | hour | minute | none ] ] | MissingTerminationCharacter [ alert
[ above [ highthresh-action [ [ all | lowthresh-action ] | email | fence | none | raslog ] | lowthresh-action [ all | email none |
raslog ] | below [ highthresh-action [ all | email | fence | none raslog ] | lowthresh-action [ all | email | none | raslog ] ] |
threshold [ buffer | high-threshold | low-threshold | timebase [ day | hour | minute | none ] ] ] ] ] }
```

no threshold-monitor interface

## Parameters

### apply

Applies a custom policy that has been created by the policy operand.

### pause

Pause monitoring.

### policy

Specifies a policy name for monitoring by means of custom settings, rather than default settings. A policy name is required before additional configurations can be made. This operation is not supported from a secondary node.

### WORD

Specifies the name of a custom policy configuration that can be saved and applied by means of the **apply** operand.

### type Ethernet

Enables gigabit Ethernet interface monitoring.

### area

Enables policy configuration.

### CRCAlignErrors

The total number of frames received with either a bad Frame Check Sequence (FCS) or an alignment error.

### IFG

The minimum-length interframe gap (IFG) between successive frames is violated. The typical minimum IFG is 12 bytes.

### MissingTerminationCharacter

The number of frames that terminate in anything other than the Terminate character.

**SymbolErrors**

The number of words received as an unknown (invalid) symbol. Large symbol errors indicate a bad device, cable, or hardware.

**alert**

Specifies whether an alert is sent when a threshold value is either above or below a threshold trigger.

**above**

Enables setting a value for **highthresh-action**, which specifies the action to be taken when a high threshold is exceeded.

**below**

Enables setting a value for **highthresh-action** and **lowthresh-action**, which specifies the action to be taken when a low threshold is exceeded.

**all**

Specifies that email and RASLog messaging are used, and that Port Fencing is applied in the case of highthresh-action only.

**email**

Specifies that an email message is sent.

**fence**

Specifies that Port Fencing is applied, which disables the port until further action is taken. This is available only for **highthresh-action**.

**none**

Specifies that no alert notification or other action (Port Fencing) is taken.

**raslog**

Specifies RASLog messaging.

**limit**

Specifies the percent of threshold usage, from 0 through 80. The default is 75.

**poll**

Specifies the polling interval in seconds, from 0 through 3600. The default is 120.

**retry**

Specifies the number of polling retries before desired action is taken, from 1 through 100. The default is 3.

**threshold**

Specifies the values for high, low, buffer, and timebase thresholds. These values are used to trigger different alerts and Port Fencing.

**buffer**

An integer value.

**high-threshold**

An integer value.

**low-threshold**

An integer value.

**timebase**

Calculates differences between current and previous data taken over a variety of intervals, for comparison against the preset threshold boundary.

- day**  
Calculates the difference between a current data value and that value a day ago.
- hour**  
Calculates the difference between a current data value and that value an hour ago.
- minute**  
Calculates the difference between a current data value and that value a minute ago.
- none**  
Compares a data value to a threshold boundary level.

## Modes

RBridge ID configuration mode

## Usage Guidelines

When any monitored error crosses the configured high or low threshold, an alert is generated and a problem port can be taken out of service. Use this command to monitor port statistics on all external gigabit Ethernet interfaces and generate a variety of actions, from alerts through Port Fencing.

## Examples

Typical multiple threshold example with fencing.

```
switch(config-rbridge-id-154)# threshold-monitor interface policy mypolicy type Ethernet area IFG alert  
above highthresh-action fence raslog lowthresh-action email raslog
```

## Related Commands

[rbridge-id](#)

# threshold-monitor memory

Configures monitoring of the memory usage of the system and alerts the user when configured thresholds are exceeded.

## Syntax

```
threshold-monitor memory { [ actions [ none | raslog { high-limit percent | limit percent | low-limit percent | poll
polling_interval | retry number_of_retries } | high-limit percent | limit percent | low-limit percent | poll polling_interval | retry
number_of_retries ] ] }
```

```
no threshold-monitor memory
```

## Parameters

### actions

Specifies the action to be taken when a threshold is exceeded.

#### *none*

No action is taken. This is the default.

### raslog

Specifies RASLog messaging.

### high-limit

Specifies an upper limit for memory usage as a percentage of available memory.

#### *percent*

This value must be greater than the value set by **limit** . When memory usage exceeds this limit, a RASLog CRITICAL message is sent. Values range from 0 through 80 percent. The default is 70 percent.

### limit

Specifies the baseline memory usage limit as a percentage of available resources.

#### *percent*

When this value is exceeded, a RASLog WARNING message is sent. When the usage returns below the value set by **limit** , a RASLog INFO message is sent. Values range from 0 through 80 percent. The default is 60 percent.

### low-limit

Specifies a lower limit for memory usage as percentage of available memory.

#### *percent*

This value must be smaller than the value set by **limit** . When memory usage exceeds or falls below this limit, a RASLog INFO message is sent. The default is 40 percent.

### poll

Specifies the polling interval in seconds.

#### *polling\_interval*

The range is from 0 through 3600. The default is 120

### retry

Specifies the number of polling retries before desired action is taken.



*number\_of\_retries*

Range is from 1 through 100. The default is 3.

## Modes

RBridge ID configuration mode

## Examples

```
device(config-rbridge-id-154)# threshold-monitor memory actions none high-limit 80 low-limit 50 limit  
70 retry 2 poll 30
```

## Related Commands

[rbridge-id](#)

# threshold-monitor security

Configures monitoring of security parameters, such as Telnet and login violations.

## Syntax

```
threshold-monitor security { [ apply policy_name | pause | policy policy_name ] area [ login-violation [ alert [ above
[ highthresh-action [ [ all | lowthresh-action ] | email | fence | none | raslog ] | lowthresh-action [ all | email none | raslog ] |
below [ highthresh-action [ all | email | fence | none raslog ] | lowthresh-action [ all | email | none | raslog ] ] | threshold
[ buffer | high-threshold | low-threshold | timebase [ day | hour | minute | none ] ] | telnet-violation [ alert [ above
[ highthresh-action [ [ all | lowthresh-action ] | email | fence | none | raslog ] | lowthresh-action [ all | email none | raslog ] |
below [ highthresh-action [ all | email | fence | none raslog ] | lowthresh-action [ all | email | none | raslog ] ] | threshold
[ buffer | high-threshold | low-threshold | timebase [ day | hour | minute | none ] ] ] ] ] }
```

```
no threshold-monitor security
```

## Command Default

For other security monitoring defaults, see the "System Monitor" chapter in the *Network OS Security Configuration Guide*.

## Parameters

### apply

Applies a custom policy that has been created by the **policy** operand.

*policy\_name*

Name of a custom policy configuration created by the **policy** operand.

### pause

Pauses monitoring.

### policy

Specifies a policy name for monitoring by means of custom settings, rather than default settings. A policy name is required before additional configurations can be made. This operation is not supported from a secondary node.

*policy\_name*

Name of a custom policy configuration that can be saved and applied by means of the **apply** operand.

### area

Enables policy configuration.

### login-violation

Enables monitoring of login violations.

### alert

Specifies whether an alert is sent when a threshold value is either above or below a threshold trigger.

### above

Enables setting a value for **highthresh-action**, which specifies the action to be taken when a high threshold is exceeded.

**below**  
Enables setting a value for **highthresh-action** and **lowthresh-action**, which specifies the action to be taken when a low threshold is exceeded.

**all**  
Specifies that email and RASLog messaging are used, and that Port Fencing is applied in the case of **highthresh-action** only.

**all**  
Specifies that email and RASLog messaging are used.

**email**  
Specifies that an email message is sent.

**fence**  
Specifies that Port Fencing is applied, which disables the port until further action is taken.

**none**  
No alert is sent

**raslog**  
Specifies RASLog messaging.

**limit**  
Specifies the percent of threshold usage, from 0 through 80. The default is 75.

**poll**  
Specifies the polling interval in seconds, from 0 through 3600. The default is 120.

**retry**  
Specifies the number of polling retries before desired action is taken, from 1 through 100. The default is 3.

**threshold**  
Specifies the values for high, low, buffer, and timebase thresholds. These values are used to trigger different alerts and Port Fencing.

**buffer**  
An integer value.

**high-threshold**  
An integer value.

**low-threshold**  
An integer value.

**timebase**  
Calculates differences between current and previous data taken over a variety of intervals, for comparison against the preset threshold boundary.

**day**  
Calculates the difference between a current data value and that value a day ago.

**hour**  
Calculates the difference between a current data value and that value an hour ago.

**minute**  
Calculates the difference between a current data value and that value a minute ago.

**none**

Compares a data value to a threshold boundary level.

**telnet-violation**

Enables monitoring of Telnet violations. Operands are as for **login-violation** .

## Modes

RBridge ID configuration mode

## Examples

Here are examples of typical commands:

```
switch(config-rbridge-id-154)# threshold-monitor security policy mypolicy area telnet-violation
threshold high-threshold 10 buffer 3
```

```
switch(config-rbridge-id-154)# threshold-monitor security policy mypolicy area login-violation timebase
hour
```

## Related Commands

[rbridge-id](#)

# threshold-monitor sfp

Configures monitoring of SFP parameters.

## Syntax

```
threshold-monitor sfp { [ apply policy_name | pause | policy policy_name ] type SFP_type area parameters alert [ above
  [ highthresh-action [ [ all | lowthresh-action ] | email | none | raslog ] | lowthresh-action [ all | email none | raslog ] | below
  [ highthresh-action [ all | email | none raslog ] | lowthresh-action [ all | email | none | raslog ] ] | threshold [ buffer | high-
  threshold | low-threshold | timebase [ day | hour | minute | none ] ] ] }
```

```
no threshold-monitor sfp
```

## Command Default

By default, SFP is not monitored.

## Parameters

**apply** *policy\_name*

Applies a custom policy that has been created by the **policy** operand.

**pause**

Pause monitoring.

**policy**

Specifies a policy name for monitoring by means of custom settings, rather than default settings. A policy name is required before additional configurations can be made. This operation is not supported from a secondary node.

*policy\_name*

Name of a custom policy configuration that can be saved and applied by means of the **apply** operand.

**type**

Specifies the SFP type. Possible completions are as follows:

**1GLR**

– SFP Type 1GLR

**1GSR**

– SFP Type 1GSR

**10GLR**

– SFP Type 10GLR

**10GSR**

– SFP Type 10GSR

**10GUSR**

– SFP Type 10GUSR

**100GSR**

– SFP Type 100GSR

**QSFP**

– SFP type QSFP

**area**

Specifies one of the following SFP parameters to be monitored. See Defaults, below.

**Current**

Measures the current supplied to the SFP transceiver.

**RXP**

Measures the incoming laser power, in microWatts ( $\mu$ W).

**TXP**

Measures the outgoing laser power, in  $\mu$ W).

**Temperature**

Measures the temperature of the SFP, in degrees Celsius.

**Voltage**

Measures the voltage supplied to the SFP.

**alert**

Specifies whether an alert is sent when a threshold value is either above or below a threshold trigger.

**above**

Enables setting a value for **highthresh-action**, which specifies the action to be taken when a high threshold is exceeded.

**below**

Enables setting a value for **highthresh-action** and **lowthresh-action**, which specifies the action to be taken when a low threshold is exceeded.

**all**

Specifies that email and RASLog messaging are used, and that Port Fencing is applied in the case of **highthresh-action** only.

**all**

Specifies that email and RASLog messaging are used.

**email**

Specifies that an email message is sent.

**none**

Specifies that no alert is sent.

**raslog**

Specifies RASLog messaging.

**limit**

Specifies the percent of threshold usage, from 0 through 80. The default is 75.

**poll**

Specifies the polling interval in seconds, from 0 through 3600. The default is 120.

**retry**

Specifies the number of polling retries before desired action is taken, from 1 through 100. The default is 3.

**threshold**

Specifies the values for high, low, buffer, and timebase thresholds. These values are used to trigger different alerts and Port Fencing.

**buffer**

An integer value.

**high-threshold**

An integer value.

**low-threshold**

An integer value.

**timebase**

Calculates differences between current and previous data taken over a variety of intervals, for comparison against the preset threshold boundary.

**day**

Calculates the difference between a current data value and that value a day ago.

**hour**

Calculates the difference between a current data value and that value an hour ago.

**minute**

Calculates the difference between a current data value and that value a minute ago.

**none**

Compares a data value to a threshold boundary level.

## Modes

RBridge ID configuration mode

## Examples

A typical command might look like this:

```
device(config)# threshold-monitor sfp custom type QSFP area rxp threshold high-threshold 2000 low-threshold 1000
```

## Related Commands

[rbridge-id](#)

# timeout fnm

Under Access Gateway, sets the fabric name monitoring time-out value (TOV) for Modified Managed Fabric Name Monitoring (M-MFNM) mode.

## Syntax

`timeout fnm value`

## Parameters

*value*

Specifies a value from 30 to 3600 seconds. The default is 120 seconds.

## Modes

Access Gateway configuration mode

## Usage Guidelines

This command sets the time out value (TOV) for M-MFNM queries of the fabric name to detect whether all N\_Ports in a port group are physically connected to the same physical or virtual fabric. (M-MFNM is a port-grouping mode that prevents connections from the AG VDX switch to multiple SANs.)

## Examples

The following example sets the fabric name monitoring TOV value.

```
device# configure terminal
device(config)# rbridge 3
device(config-rbridge-id-3)# ag
device(config-rbridge-id-3-ag)# timeout fnm 60
```

## History

Release version	Command history
6.0.1	This command is available only as an independent command, in AG configuration mode. Previously, in RBridge-ID configuration mode it could be executed with the <b>ag</b> prefix.

## Related Commands

[ag](#)



# timers

Configures Link State Advertisement (LSA) pacing and Shortest Path First (SPF) throttle timers.

## Syntax

```
timers { lsa-group-pacing interval | throttle spf start hold max }
```

## Command Default

Enabled.

## Parameters

### lsa-group-pacing *interval*

Specifies the interval at which OSPF LSAs are collected into a group and refreshed, check-summed, or aged by the OSPF process. Valid values range from 10 to 1800 seconds. The default is 240 seconds.

### throttle spf

Specifies start, hold and maximum wait intervals for throttling SPF calculations for performance. The values you enter are in milliseconds.

#### *start*

Initial SPF calculation delay. Valid values range from 0 to 60000 milliseconds. The default is 0 milliseconds.

#### *hold*

Minimum hold time between two consecutive SPF calculations. Valid values range from 0 to 60000 milliseconds. The default is 5000 milliseconds.

#### *max*

Maximum wait time between two consecutive SPF calculations. Valid values range from 0 to 60000 milliseconds. The default is 10000 milliseconds.

## Modes

OSPF router configuration mode

OSPF VRF router configuration mode

## Usage Guidelines

The device paces LSA refreshes by delaying the refreshes for a specified time interval instead of performing a refresh each time an individual LSA refresh timer expires. The accumulated LSAs constitute a group, which the device refreshes and sends out together in one or more packets.

The LSA pacing interval is inversely proportional to the number of LSAs the device is refreshing and aging. For example, if you have a large database of 10,000 LSAs, decreasing the pacing interval enhances performance. If you have a small database of about 100 LSAs, increasing the pacing interval to 10 to 20 minutes may enhance performance.

Enter the **no timers lsa-group-pacing** to restore the pacing interval to its default value.

Enter **no timers throttle spf** to set the SPF timers back to their defaults.

## Examples

The following example sets the LSA group pacing interval to 30 seconds.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# timers lsa-group-pacing 30
```

The following example sets the SPF delay to 10000 milliseconds, the hold time to 15000 milliseconds, and the maximum wait time to 30000 milliseconds.

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf
device(config-router-ospf-vrf-default-vrf)# timers throttle spf 10000 15000 30000
```

# timers (BGP)

Adjusts the interval at which BGP KEEPALIVE and HOLDTIME messages are sent.

## Syntax

```
timers { keep-alive keepalive_interval hold-time holdtime_interval }
```

```
no timers
```

## Parameters

**keep-alive** *keepalive\_interval*

Frequency in seconds with which a device sends keepalive messages to a peer. Range is from 0 through 65535 seconds. The default is 60.

**hold-time** *holdtime\_interval*

Interval in seconds that a device waits to receive a keepalive message from a peer before declaring that peer dead. Range is from 0 through 65535 seconds. The default is 180.

## Modes

BGP configuration mode

## Usage Guidelines

The KEEPALIVE and HOLDTIME message interval is overwritten when the **fast-external-failover** command takes effect on a down link to a peer.

You must enter a value for **keep-alive** before you can enter a value for **hold-time**. Both values must be entered. If you only want to adjust the value of one parameter, enter the default value of the parameter that you do not want to adjust.

The **no** form of the command clears the timers.

## Examples

The following example sets the keepalive timer for a device to 120 seconds and the hold-timer to 360 seconds.

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# timers keep-alive 120 hold-time 360
```

## Related Commands

[fast-external-fallover](#)

# timers (OSPFv3)

Configures Link State Advertisement (LSA) pacing and Shortest Path First (SPF) timers.

## Syntax

```
timers { lsa-group-pacing interval | spf start hold }
```

## Command Default

Enabled.

## Parameters

### lsa-group-pacing *interval*

Specifies the interval at which OSPFv3 LSAs are collected into a group and refreshed, check-summed, or aged by the OSPFv3 process. Valid values range from 10 to 1800 seconds. The default is 240 seconds.

### spf

Specifies start and hold intervals for SPF calculations for performance. The values you enter are in milliseconds.

#### *start*

Initial SPF calculation delay. Valid values range from 0 to 65535 seconds. The default is 5 seconds.

#### *hold*

Minimum hold time between two consecutive SPF calculations. Valid values range from 0 to 65535 seconds. The default is 10 milliseconds.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The device paces LSA refreshes by delaying the refreshes for a specified time interval instead of performing a refresh each time an individual LSA refresh timer expires. The accumulated LSAs constitute a group, which the device refreshes and sends out together in one or more packets.

The LSA pacing interval is inversely proportional to the number of LSAs the device is refreshing and aging. For example, if you have a large database of 10,000 LSAs, decreasing the pacing interval enhances performance. If you have a small database of about 100 LSAs, increasing the pacing interval to 10 to 20 minutes may enhance performance.

The **no timers lsa-group-pacing** command restores the pacing interval to its default value.

The **no timers spf** command sets the SPF timers back to their defaults.

## Examples

The following example sets the LSA group pacing interval to 30 seconds.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# timers lsa-group-pacing 30
```

The following example sets the SPF delay time to 10 and the hold time to 20.

```
device# configure terminal
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 router ospf
device(config-ipv6-router-ospf-vrf-default-vrf)# timers spf 10 20
```

## History

Release version	Command history
5.0.0	This command was introduced.

# traceroute

Traces the network path of packets as they are forwarded to a destination address.

## Syntax

```
traceroute { IPv4_address | host-name | ipv6 [ dest-ipv6-addr | host-name ] } [ maxttl value ] [ minttl value ] [ src-addr src-addr ] [ timeout seconds ] [ vrf vrf-name ]
```

## Parameters

*IPv4\_address*

Specifies the IPv4 address of the destination device.

*host-name*

Specifies the hostname of the destination device.

**ipv6** *dest-ipv6-addr*

Specifies the IPv6 address of the destination device. This parameter is valid only with the **ping** command.

**maxttl** *value*

Maximum Time To Live value in a number of hops.

**minttl** *value*

Minimum Time To Live value in a number of hops.

**src-addr** *address*

Specifies the IPv4 or IPv6 address of the source device.

**timeout** *seconds*

The traceroute timeout value.

**vrf** *vrf-name*

Name of the VRF. If no VRF is specified, the default-vrf is used.

## Modes

Privileged EXEC mode

## Usage Guidelines

To use the **traceroute** command on the management VRF, enter **mgmt-vrf** as follows. You must enter the name of the management VRF manually.

```
device# traceroute 1.1.1.1 vrf mgmt-vrf
```

## Examples

The following example executes an IPv4 traceroute.

```
device# traceroute 172.16.4.80

traceroute to 172.16.4.80 (172.16.4.80), 64 hops max
 1  10.24.80.1 (10.24.80.1) 0.588ms 0.139ms 0.527ms
 2  10.31.20.61 (10.31.20.61) 0.550ms 0.254ms 0.234ms
 3  10.16.200.113 (10.16.200.113) 0.408ms 0.285ms 0.282ms
 4  10.110.111.202 (10.110.111.202) 5.649ms 0.283ms 0.288ms
 5  10.130.111.38 (10.130.111.38) 1.108ms 0.712ms 0.704ms
 6  10.192.0.42 (10.192.0.42) 37.053ms 32.985ms 41.744ms
 7  172.16.56.10 (172.16.56.10) 33.110ms 33.349ms 33.114ms
 8  172.16.4.9 (172.16.4.9) 34.096ms 33.023ms 33.122ms
 9  172.16.4.80 (172.16.4.80) 76.702ms 83.293ms 79.570ms
```

The following example executes an IPv6 traceroute, with minimum and maximum TTL values.

```
device# traceroute ipv6 fec0:60:69bc:92:218:8bff:fe40:1470 maxttl 128 minttl 30 src-addr fec0:60:69bc:
92:205:33ff:fe9e:3f20 timeout 3

traceroute to fec0:60:69bc:92:218:8bff:fe40:1470 (fec0:60:69bc:92:218:8bff:fe40:1470), 128 hops max, 80
byte packets
30 fec0:60:69bc:92:218:8bff:fe40:1470 (fec0:60:69bc:92:218:8bff:fe40:1470) 2.145 ms 2.118 ms 2.085
ms
```

## Related Commands

[ping](#)

# track (Fabric-Virtual-Gateway)

Tracks an interface, network, or next hop.

## Syntax

```
track interface {<N> gigabitethernet rbridge-id/slot/port | port-channel number } priority range
no track interface {<N> gigabitethernet rbridge-id/slot/port | port-channel number } priority range
track network A.B.C.D/mask priority range
no track network A.B.C.D/mask
track next-hop ip-address priority range
no track next-hop ip-address
```

## Command Default

None

## Parameters

### interface

Interface type.

### network

Network address

### next-hop

Next hop IP address

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace<N>**gigabitethernet** with the desired operand (for example, tengigabitethernet specifies a 10-Gb Ethernet port). The use of **gigabitethernet** without a speed value specifies a 1-Gb Ethernet port.

#### rbridge-id

Specifies an RBridge ID.

#### slot

Specifies a valid slot number.

#### port

Specifies a valid port number.

### port-channel number

Specifies the port-channel number. Valid values range from 1 through 6144.

### priority range

The track priority range is from 1 through 254.

### A.B.C.D/mask

Network address in A.B.C.D/mask format.



*ip-address*  
IP address.

## Modes

Fabric-virtual-gateway in an RBridge VE interface IPv4 or IPv6 configuration mode

## Usage Guidelines

Enter the **no** form of the command with the specified interface to remove the tracked port configuration.

## Examples

The following example shows how to track a 40-gigabit Ethernet interface with a priority of 34.

```
device(config)# rbridge-id 55
device(config-rbridge-id-55)# interface ve 1
device(config-Ve-1)# ip fabric-virtual-gateway
device(config-ip-fabric-virtual-gw)# track interface fortygigabitethernet 55/0/51 priority 34
```

The following example shows how to track a network with a priority of 10.

```
device(config)# rbridge-id 58
device(config-rbridge-id-58)# interface ve 1
device(config-Ve-1)# ipv6 fabric-virtual-gateway
device(config-ipv6-fabric-virtual-gw)# track network 1::/64 priority 10
```

The following example shows how to track a next hop with a priority of 10.

```
device(config)# rbridge-id 58
device(config-rbridge-id-58)# interface ve 1
device(config-Ve-1)# ipv6 fabric-virtual-gateway
device(config-ipv6-fabric-virtual-gw)# track next-hop 10.1.1.101 priority 10
```

## History

Release version	Command history
5.0.1	This command was introduced.

# track (LST)

Enables and configures link-state tracking (LST), which prevents traffic loss between upstream and downstream links. LST is effective for redundant-link topology.

## Syntax

```
track { interface { ethernet rbridge-id/slot/port | port-channel number } | min-link number | enable | remove all }
no track interface { ethernet rbridge-id/slot/port | port-channel number }
no track enable
no track min-link
```

## Command Default

By default, LST is disabled.

## Parameters

### interface

On a downlink interface, configures tracking for an uplink interface.

**ethernet** *rbridge-id/slot/port*

Specifies a physical interface.

*rbridge-id*

Specifies an RBridge ID.

*slot*

Specifies a valid slot number.

*port*

Specifies a valid port number.

**port-channel** *number*

Specifies a port-channel interface.

**min-link** *number*

For multiple-uplink topologies, specifies the minimum number of available uplinks below which LST shuts down the downlinks. Acceptable values are 1 through 24.

**enable**

Enables LST, according to the specified parameters.

**remove all**

Removes all uplinks from the downlink interface that you are configuring.

## Modes

One of the following interface configuration modes:

- One of the supported **<N>gigabitethernet** modes

- Port-channel

## Usage Guidelines

You can implement LST on multiple hops in a network.

LST operates on the operational level of uplinks rather than their STP forwarding state. So the following are required:

- The redundant network must be loop-free.
- If STP or RSTP is enabled, every uplink under LST must be in STP forwarding state.

LST is supported only for the following interface types:

- Supported physical ethernet ports (including breakout ports), but not FC ports.
- Port-channels. However, port-channels are not supported for LST under FlexPort.

In a VCS cluster, local RBridge uplinks and downlinks are supported for LST; remote ports are not supported.

(Available only if LST is currently enabled on an interface) To disable link-state tracking of a specific uplink, enter the **no track interface** command on the downlink interface.

(Available only if LST is currently enabled on an interface) To disable link-state tracking, use the **no track enable** command.

(Available only if **min-link** is specified on an interface) To restore the default setting of no minimum number of links, use the **no track min-link** command.

## Examples

The following example configures and enables LST on an interface (independent RBridge or VCS principal RBridge) with one uplink.

```
device# configure terminal
device(config)# interface tengigabitethernet 1/0/1
device(conf-if-te-1/0/1)# track interface ethernet 1/0/20
device(conf-if-te-1/0/1)# track enable
```

The following example configures and enables LST on an interface (independent RBridge or VCS principal RBridge) with multiple uplinks and **min-link = 1**.

```
device# configure terminal
device(config)# interface tengigabitethernet 3/0/8
device(conf-if-te-3/0/8)# track interface ethernet 3/0/18
device(conf-if-te-3/0/8)# track interface ethernet 3/0/19
device(conf-if-te-3/0/8)# track min-link 1
device(conf-if-te-3/0/8)# track enable
```

## History

Release version	Command history
6.0.1	This command was introduced.

## Related Commands

[show interface](#), [show running-config interface tengigabitethernet](#), [show track summary](#)

# track (VRRP)

Enables VRRP tracking for a specified interface. VRRP Extended (VRRP-E) sessions can track a specified interface or a network.

## Syntax

```
track { <N>gigabitethernet rbridge-id/slot/port | port-channel number } [ priority value ]
track network { ip-address/mask | ipv6-address/mask } [ priority value ]
no track { <N>gigabitethernet rbridge-id/slot/port | port-channel number } [ priority value ]
no track network { ip-address/mask | ipv6-address/mask } [ priority value ]
```

## Command Default

The default priority value is 2.

## Parameters

### <N>gigabitethernet

Represents a valid, physical Ethernet subtype for all available Ethernet speeds. Enter ? to see which interface subtypes are available. Replace <N>gigabitethernet with the desired operand (for example, tengigabitethernet specifies a 10-Gb Ethernet port). The use of gigabitethernet without a speed value specifies a 1-Gb Ethernet port.

#### rbridge-id

Specifies an RBridge ID.

#### slot

Specifies a valid slot number.

#### port

Specifies a valid port number.

### port-channel number

Specifies the port-channel number. Valid values range from 1 through 6144.

### priority value

The track priority is a number from 1 through 254, and is used when a tracked interface or network up or down event is detected. For VRRP, if the tracked interface goes offline, the specified priority value is subtracted from the priority of the current device. For VRRP-E, if the tracked interface or network goes offline, the current device priority is reduced by the configured priority value. If the tracked interface or network comes online, the specified priority value is added to the priority of the current device.

### network

Enables tracking of a specified network. Network tracking is supported only on VRRP-E sessions.

#### ip-address

Specifies an IPv4 network address.

#### ipv6-address

Specifies an IPv6 network address.

*mask*

Specifies a mask for the associated IP or IPv6 subnet.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

This command can be used to track interfaces for VRRP or VRRP-E. Only VRRP-E sessions support network tracking.

For VRRP, the tracked interface can be any 10-gigabit Ethernet, 40-gigabit Ethernet, 1-gigabit Ethernet, or port-channel interface other than the one on which this command is issued.

The networks to be tracked can be either present or absent from the Routing Information Base (RIB).

The maximum number of interfaces or networks you can track per virtual router is 16.

Enter **no track** with the specified interface or network to remove the tracked port or tracked network configuration.

## Examples

To set the track port to 21/2/4 and the track priority to 60:

```
device(config)# rbridge-id 21
device(config-rbridge-id-21)# protocol vrrp
device(config-rbridge-id-21)# int te 21/1/6
device(config-if-te-21/1/6)# vrrp-group 1
device(config-vrrp-group-1)# track tengigabitethernet 21/2/4 priority 60
```

The following example shows how to configure network 10.1.1.0/24 to be tracked, and if the network goes down, the VRRP-E device priority is lowered by a value of 20. The lower priority may trigger a switchover and a backup device with a higher priority becomes the new master for VRRP-E group 1.

```
device(config)# rbridge-id 1
device(config-rbridge-id-1)# protocol vrrp-extended
device(config-rbridge-id-1)# interface ve 100
device(config-Ve-100)# vrrp-extended-group 1
device(config-vrrp-group-1)# track network 10.1.1.0/24 priority 20
```

## History

Release version	Command history
6.0.1	This command was modified to add the <b>network</b> keyword to allow tracking of networks for VRRP-E sessions.

## Related Commands

[vrrp-group](#)

# transmit-holdcount

Configures the maximum number of Bridge Protocol Data Units (BPDUs) transmitted per second for the Multiple Spanning Tree Protocol (MSTP), Rapid Spanning Tree Protocol (RSTP), and R-PVST+.

## Syntax

`transmit-holdcount` *number*

`no transmit-holdcount`

## Command Default

6 units

## Parameters

*number*

Specifies the number of BPDUs than can be sent before pausing for 1 second. Valid unit values range from 1 through 10.

## Modes

Protocol Spanning Tree MSTP configuration mode

## Usage Guidelines

Extreme Network OS supports PVST+ and R-PVST+only. The PVST and R-PVST protocols are proprietary to Cisco and are not supported.

Enter `no transmit-holdcount` to return to the default setting.

## Examples

To change the number of BPDUs transmitted to 3 units:

```
device(conf-mstp) # transmit-holdcount 3
```

## Related Commands

[show spanning-tree mst detail](#)

# transport-service

In a Virtual Fabrics context, associates a service VF with a trunk port interface as a transport VF.

## Syntax

```
transport-service tsid  
no transport-service tsid
```

## Command Default

This feature is disabled by default.

## Parameters

*tsid*  
The transport LAN service ID. Range is from 1 through 1000.

## Modes

Interface subtype configuration mode

## Usage Guidelines

In a Virtual Fabrics context, use this command to associate a service VF (VLAN ID > 4095, through 8191) to a trunk port interface as a transport VF.

This command does not apply to standard (802.1Q) VLANs (VLAN IDs from 1 through 4095).

This command is not supported when issued from a secondary node.

Enter **no transport-service** *tsid* to remove the service VF from the trunk port interface as a transport VF.

## Examples

Configure a classified VLAN and assign it to transport VF instance 10:

```
switch(config)# interface vlan 5000  
switch(config-vlan-5000)# transport-service 10
```

## Related Commands

[interface vlan](#), [vcs virtual-fabric enable](#)

# trigger

Defines event-handler triggers. When the trigger-condition occurs, a Python script is run.

## Syntax

```
trigger trigger-id { raslog raslog-id [ pattern posix-ext-regex ] | vcs switch-event }
no trigger [ trigger-id ]
```

## Command Default

No trigger is defined.

## Parameters

*trigger-id*

Specifies an ID number for the trigger. Valid values are 1 through 100, and must be unique per event-handler profile.

**raslog** *raslog-id*

Specifies a RASlog message ID as the trigger. String can be 1 through 32 characters long.

**pattern** *posix-ext-regex*

Specifies a POSIX extended regular expression to search within the specified RASlog message ID.

**vcs** *switch-event*

Specifies a switch event as the trigger. Valid *switch-event* values are as follows:

**switch-bootup**

The switch booted and boot-time configuration is applied.

**switch-ready-for-configuration**

The switch is ready to receive a configuration through an event-handler action.

## Modes

Event-handler configuration mode

## Usage Guidelines

You can create from 1 through 100 triggers per profile, but all must be of the same type (**raslog** or **vcs**).

You can also define one trigger as part of the **event-handler** command.

To delete one or all triggers, use the **no** form of this command, as follows:

- To delete all triggers, enter **no trigger**.
- To delete a specific trigger, enter **no trigger** *trigger-id*

### NOTE

You cannot delete the last remaining trigger from an activated event-handler profile.



You can modify an existing trigger without deleting it and then re-creating it.

If the event-handler for which you are modifying triggers is active on one or more RBridges, the changes take effect with no need to de-activate and re-activate the event-handler.

A Python event-handler script runs only if all of the following occur:

- Using the **copy** command, copy the Python file to the `flash://` location on the device.
- Using the **event-handler** command, create an event-handler profile.
- In configuration mode for that profile:
  - Using the **trigger** command, create one or more triggers.
  - Using the **action** command, specify the Python script that will be triggered.
- Using the **event-handler activate** command, activate an instance of the event handler.
- The trigger event occurs.

## Examples

The following example defines triggers in two event handlers.

```
device# configure terminal
device(config)# event-handler eventHandler1
device(config-event-handler-eventHandler1)# trigger 1 vcs switch-bootup
device(config-event-handler-eventHandler1)# event-handler eventHandler2
device(config-event-handler-eventHandler2)# trigger 1 raslog VCS-1003
device(config-event-handler-eventHandler2)# trigger 2 raslog VCS-1004
```

The following example defines a trigger that uses POSIX extended REGEX to search for a match within a specified RASlog message ID.

```
device# configure terminal
device(config-event-handler-eventHandler1)# event-handler eventHandler2
device(config-event-handler-eventHandler2)# trigger 1 raslog NSM-1003 pattern Interface (One|Ten|Forty|
Hundred)GigabitEthernet 89/0/[1-9] is link down
```

RASlog message NSM-1003 includes "**interface** *interface-name* is link down", indicating that an interface is offline because the link is down. The REGEX searches within such a message for an interface from 89/0/1 through 89/0/9.

## History

Release version	Command history
6.0.1	This command was introduced.
7.1.0	The command was modified to support POSIX extended regular expressions to search in a specified RASlog message ID.

# trigger-function

For an implementation of an event-handler profile, if multiple triggers are defined for an event-handler action, specifies if the action runs only if all of the triggers occur; or if one is sufficient.

## Syntax

```
trigger-function { OR | AND { time-window seconds } }
```

```
no trigger-function
```

## Command Default

The event-handler action runs if any of the triggers occur.

## Parameters

**OR**

The event-handler action runs if any of the triggers occur.

**AND**

The event-handler action runs only if all of the triggers occur.

**time-window seconds**

In seconds, specify the time window within which all of the triggers must occur in order that the event-handler action runs.

Following an initial triggering of an event-handler action, any subsequent trigger launches the action an additional time if the following conditions are true:

- The **trigger-mode** parameter is set to the default **each-instance**.
- The subsequent trigger occurs within the specified **time-window**.

## Modes

Event-handler activation mode

## Usage Guidelines

The **no** form of this command sets the **trigger-function** setting to the default **OR** option.

## Examples

The following example determines that the event-handler action runs only if all of the triggers occur within 120 seconds.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# trigger-function AND time-window 120
```

The following example resets **trigger-function** to the default **OR** option.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# no trigger-function
```

## History

Release version	Command history
6.0.1	This command was introduced.

# trigger-mode

For an implementation of an event-handler profile, specifies if recurring trigger conditions can launch an event-handler action more than once.

## Syntax

`trigger-mode mode`

`no trigger-mode`

## Command Default

Each time the trigger condition occurs, the event-handler action is launched.

## Parameters

*mode*

Specifies if an event-handler action can be triggered only once or more than once.

**each-instance**

The event-handler action is launched on each trigger instance received.

**on-first-instance**

As long as the device is running, the event-handler action is launched only once. Following a device restart, the event-handler action can be triggered again.

**only-once**

For the duration of a device configuration, the event-handler action is launched only once.

## Modes

Event-handler activation mode

## Usage Guidelines

The `no` form of this command resets the `trigger-mode` setting to the default `each-instance` option.

## Examples

The following example sets the trigger mode to **on-first-instance**.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# trigger-mode on-first-instance
```

The following example resets `trigger-mode` to the default value of **each-instance**.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# event-handler activate eventHandler1
device(config-activate-eventHandler1)# no trigger-mode
```

## History

Release version	Command history
6.0.1	This command was introduced.

# trunk-enable

Enables port trunking on a Fibre Channel port.

## Syntax

**trunk-enable**

**no trunk-enable**

## Modes

Interface Fibre Channel configuration mode

## Usage Guidelines

This command can be used only on Network OS platforms with Fibre Channel ports.

A long-distance link can also be configured to be part of a trunk group.

While using R\_RDY mode flow control, an E\_Port cannot form trunk groups of long distance links even if trunking is enabled.

To disable trunking on a Fibre Channel port, enter **no trunk-enable**.

## Examples

This example enables trunking mode on a Fibre Channel port:

```
device(config)# interface FibreChannel 7/0/2
device(conf-FibreChannel-7/0/2)# trunk-enable
```

This example disables trunking mode on a Fibre Channel port:

```
device (config)# interface FibreChannel 7/0/2
device(conf-FibreChannel-7/0/2)# no trunk-enable
```

# tunable-optics

This command assigns channels to tunable optic interfaces (T-SFP+) for specific wavelengths.

## Syntax

```
tunable-optics sfpp channel channel_number
```

## Command Default

The T-SFP+ optic defaults to a "no wavelength" state before being activated.

## Modes

Interface configuration mode

## Usage Guidelines

Tunable SFP+ optics are optional hardware that can be installed in Extreme VDX 6740, VDX 6940-144S fixed 10GbE ports, and 10 G linecards with optical SFPs.

If you are installing a T-SFP+ in a 144S port, the T-SFP+ optic needs to be installed in both ends of the cable. The T-SFP+ at each end of the cable link must be configured at the same wavelength by setting them to the same channel on each device.

Failure to duplicate the channel setting may allow the link to come online, but the link behavior may be erratic.

If Extreme Network OS determines an error exceeding a specified limit, a RASLOG message event occurs and the port is taken offline.

The T-SFP+ interface defaults to a "no wavelength" state. When a supported Extreme device boots, the firmware sets the desired wavelength of the T-SFP+ optic.

When a T-SFP+ interface is installed it is very important that the interface is configured to the same channel (wavelength) at both ends. Use the **show media tunable-optic-sfpp** command to determine the currently configured channel.

T-SFP+ interfaces are tuned to specific wavelengths and frequencies using pre-defined channels.

The following tables lists the frequency and wavelength assigned to channels for tunable SFP+ optic interfaces.

**TABLE 27** Supported wavelengths and channel numbers

Channel	Frequency (THz)	Wavelength (nm)
1	191.10	1568.77
2	191.15	1568.36
3	191.20	1567.95
4	191.25	1567.54
5	191.30	1567.13
6	191.35	1566.72
7	191.40	1566.31
8	191.45	1565.90
9	191.50	1565.50

**TABLE 27** Supported wavelengths and channel numbers (continued)

Channel	Frequency (THz)	Wavelength (nm)
10	191.55	1565.09
11	191.60	1564.68
12	191.65	1564.27
13	191.70	1563.86
14	191.75	1563.45
15	191.80	1563.05
16	191.85	1562.64
17	191.90	1562.23
18	191.95	1561.83
19	192.00	1561.42
20	192.05	1561.01
21	192.10	1560.61
22	192.15	1560.20
23	192.20	1559.79
24	192.25	1559.39
25	192.30	1558.98
26	192.35	1558.58
27	192.40	1558.17
28	192.45	1557.77
29	192.50	1557.36
30	192.55	1556.96
31	192.60	1556.55
32	192.65	1556.15
33	192.70	1555.75
34	192.75	1555.34
35	192.80	1554.94
36	192.85	1554.54
37	192.90	1554.13
38	192.95	1553.73
39	193.00	1553.33
40	193.05	1552.93
41	193.10	1552.52
42	193.15	1552.12
43	193.20	1551.71
44	193.25	1551.32
45	193.30	1550.92
46	193.35	1550.52
47	193.40	1550.12
48	193.45	1549.72
49	193.50	1549.32
50	193.55	1548.91



**TABLE 27** Supported wavelengths and channel numbers (continued)

Channel	Frequency (THz)	Wavelength (nm)
51	193.60	1548.51
52	193.65	1548.11
53	193.70	1547.72
54	193.75	1547.32
55	193.80	1546.92
56	193.85	1546.52
57	193.90	1546.12
58	193.95	1545.72
59	194.00	1545.32
60	194.05	1544.92
61	194.10	1544.53
62	194.15	1544.13
63	194.20	1543.73
64	194.25	1543.33
65	194.30	1542.94
66	194.35	1542.54
67	194.40	1542.14
68	194.45	1541.75
69	194.50	1541.35
70	194.55	1540.95
71	194.60	1540.56
72	194.65	1540.16
73	194.70	1539.77
74	194.75	1539.37
75	194.80	1538.98
76	194.85	1538.58
77	194.90	1538.19
78	194.95	1537.79
79	195.00	1537.40
80	195.05	1537.00
81	195.10	1536.61
82	195.15	1536.22
83	195.20	1535.82
84	195.25	1535.43
85	195.30	1535.04
86	195.35	1534.64
87	195.40	1534.25
88	195.45	1533.86
89	195.50	1533.47
90	195.55	1533.07
91	195.60	1532.68

TABLE 27 Supported wavelengths and channel numbers (continued)

Channel	Frequency (THz)	Wavelength (nm)
92	195.65	1532.29
93	195.70	1531.90
94	195.75	1531.51
95	195.80	1531.12
96	195.85	1530.72
97	195.90	1530.33
98	195.95	1529.94
99	196.00	1529.55
100	196.05	1529.16
101	196.10	1528.77
102	196.15	1528.38

## Examples

Typical command example.

```
device# configure terminal
device(config)# interface tengigabitethernet 2/0/1
device(conf-if-te-2/0/1)# tunable-optics sfpp channel 5
device(conf-if-te-2/0/1)# do show media optical-monitoring
```

N/A - Not Available.

N/S - Optical-monitoring Not Supported.

Port	Module Temperature ( C )	Supply Voltage ( mVolts )	Channel TX Power ( uWatts )	Frequency Error ( GHz )	Wavelength Error ( nm )	Bias Current ( mAmps )	Channel RX Power ( uWatts )
Fo 1/0/52	31	3306.1	N/A	37.364 34.148 33.664 33.060	2.7 3.1 3.9 1.0		
Te 1/0/1	36	3291.6	694.4	0.0	0.000	38.550	748.8
Te 1/0/48			N/S				
Te 2/0/1	33	3317.1	685.1	0.0	0.000	37.132	914.5
Te 2/0/48			N/S				

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[show media tunable-optic-sfpp](#), [show media optical-monitoring interface](#), [show media interface](#)

# tunnel replicator bum-vlans redistribute

Redistributes the VLANs on tunnels to VMware NSX replicators.

## Syntax

```
tunnel replicator bum-vlans redistribute
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is supported on the Extreme VDX 6740 and VDX 6940 series. It is not supported on the Extreme VDX 8770 series.

This command does not have a **no** form.

Use this command as a recovery tool to redistribute broadcast/unicast/multicast (BUM) VLANs on all the NSX replicator tunnels if they are not uniformly distributed. Use the **show tunnel replicator** command to view the BUM VLAN details for each replicator tunnel.

## Examples

The following command redistributes the VLANs on all NSX replicator tunnels.

```
device# tunnel replicator bum-vlans redistribute
```

## History

Release version	Command history
7.0.1	This command was introduced. It replaces the <b>tunnel service-node bum-vlans redistribute</b> command.

# tunnel tagged-ieee-bpdu

Activates IEEE BPDU packets.

## Syntax

```
tunnel tagged-ieee-bpdu
```

```
no tunnel tagged-ieee-bpdu
```

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no tagged-ieee-bpdu-enabled** to disable this feature.

This command should only be used on edge ports.

### ATTENTION

This command should be enabled when the interface is connected to a switch that sends tagged IEEE BPDU packets.

# tunnel-traceroute

Provides underlay hop information and overlay reachability information in a overlay tunnel, for enhanced visibility of packet traversal between network virtualization edge (NVE) nodes in a Clos IP Fabrics network.

## Syntax

```
tunnel-traceroute { da DA | sa SA }
```

```
tunnel-traceroute { ip | ipv6 } { dip DIP | sip SIP | vlan VLAN }
```

```
tunnel-traceroute { interface ingress_port || l4_dest_port l4_dest_port | l4_src_port l4_src_port | protocol protocol }
```

```
no tunnel-traceroute
```

## Command Default

This feature is not enabled.

## Parameters

- da** *DA*  
Specifies a destination address.
- sa** *SA*  
Specifies a source address.
- ip**  
Specifies IPv4.
- ipv6**  
Specifies IPv6.
- dip** *DIP*  
Specifies a destination IPv4 or IPv6 address.
- sip** *SIP*  
Specifies a source IPv4 or IPv6 address.
- interface** *ingress\_port*  
Specifies an ingress port (physical or port-channel).
- protocol** *protocol*  
Specifies a protocol.
- l4\_dest\_port** *l4\_dest\_port*  
Specifies a Layer 4 destination port.
- l4\_src\_port** *l4\_src\_port*  
Specifies a Layer 4 source port.

## Modes

Privileged EXEC mode

## Usage Guidelines

This feature is supported on the VDX 6740, VDX 6940, and VDX 8770 series platforms. The VDX 8770 series can be a spine node in an IP Fabrics network.

This feature does not support chained tunnels.

If an ingress port is not provided, it is derived from the source MAC address.

### debug asic-simulation-model

Use the **no** form of this command to terminate the traceroute operation.

Use the **show debug asic-simulation-model** command to view the status of this operation.

## Examples

The following IPv4 example returns a successful tunnel termination at the egress network virtualization edge (NVE).

```
device# tunnel-traceroute sa 0001.0025.1110 da 0001.0025.1120 ipv4 sip 25.1.1.10 DIP 25.1.1.20 vlan 100
      protocol udp 14-src-port 1000 4-dest-port 2000 interface port-channel 10
Hop count  next hop          RTT1          RTT2          RTT3
1          10.10.10.10        0.345 ms     0.439 ms     0.418 ms
2          3.3.3.3            0.432 ms     0.380ms      0.402 ms
          Packet is tunnel terminated at NVE 3.3.3.3 and rbridge id 1
VNI 1000 -> VLAN 100
```

The following IPv4 example returns an unsuccessful tunnel termination at the NVE.

```
device# tunnel-traceroute sa 0001.0025.1110 da 0001.0030.1110 ipv4 sip 25.1.1.10 dip 30.1.1.10 vlan 100
Hop count  next hop          RTT1          RTT2          RTT3
1          20.20.20.20        0.395 ms     0.367 ms     0.411 ms
2          3.3.3.3            0.419 ms     0.388ms      0.456 ms
          Packet is not tunnel terminated at NVE 3.3.3.3 and rbridge id 1
```

### NOTE

The above output also occurs when nodes are configured not to respond to TTL expiry errors but Network OS switches are configured to respond to those errors with ICMP replay messages.

The following example can be used to isolate a fault in an IP Fabric with all Network OS switches.

```
device# tunnel-traceroute sa 0001.0025.1110 da 0001.0030.1110 ipv4 sip 25.1.1.10 dip 30.1.1.10 vlan 100
      protocol udp 14-src-port 1000 4-dest-port 2000 interface Te 1/0/20
Hop count  next hop          RTT1          RTT2          RTT3
1          10.10.10.10        0.395 ms     0.367 ms     0.411 ms
2          *                  *             *             *
3          *                  *             *             *
.....
.....
30         *                  *             *             *
```

The following IPv6 examples return a successful tunnel termination at the egress network virtualization edge (NVE).

```
device# tunnel-traceroute SA 0001.0025.1110 DA 0001.0025.1120 IPV6 SIP 2000::10:10 DIP 2000::10:20
VLAN 100 interface port-channel 10 protocol UDP L4-Src-Port 1000 4-dest-port 2000
Hop count   next hop           RTT1           RTT2           RTT3
1           10.10.10.10        0.345 ms      0.439 ms      0.418 ms
2           3.3.3.3            0.432 ms      0.380ms       0.402 ms
           Packet is tunnel terminated at NVE 3.3.3.3 and rbridge id 1
VNI 1000 -> VLAN 100
```

```
device# tunnel-traceroute SA 0001.0025.1110 DA 0001.0030.1110 IPV6 SIP 2000::10:10 DIP 2010::20:10 VLAN
100 interface port-channel 10 protocol UDP L4-src-port 1000 4-dest-port 2000
Hop count   next hop           RTT1           RTT2           RTT3
1           20.20.20.20        0.395 ms      0.367 ms      0.411 ms
2           3.3.3.3            0.419 ms      0.388ms       0.456 ms
           Packet is tunnel terminated at NVE 3.3.3.3 and rbridge id 1
VNI 2000 -> VLAN 200
```

## History

Release version	Command history
7.2.0	This command was introduced.

# type

Specifies whether a VXLAN overlay gateway uses NSX Controller or OpenStack integration, or Layer 2 extension.

## Syntax

```
type { hardware-vtep | layer2-extension }
```

## Command Default

NSX Controller/OpenStack integration is the default behavior.

## Parameters

### hardware-vtep

Specifies NSX Controller/OpenStack integration.

### layer2-extension

Specifies Layer 2 extension.

## Modes

VXLAN overlay gateway configuration mode

## Usage Guidelines

There is no **no** form of this command.

Note the following restrictions related to changing the type:

- To change the type, ensure that the interface is removed and the overlay gateway is not attached to any RBridge.
- If changing from **hardware-vtep** to **layer2-extension**, ensure that there are no "attach vlan" configurations, as configured by the **attach vlan** command.
- If changing from **layer2-extension** to **hardware-vtep**, ensure that no "map vlan" configurations are present, as configured by the **map vlan** command.

## Examples

To specify Layer 2 extension:

```
device# config
device(config)# overlay-gateway gateway1
device(config-overlay-gw-gateway1)# type layer2-extension
```



## History

Release version	Command history
7.0.1	This command was modified to deprecate the <b>nsx</b> keyword and replace it with the <b>hardware-vtep</b> keyword, supporting both NSX Controller and OpenStack deployments.

## Related Commands

[attach vlan](#), [map vlan](#), [overlay-gateway](#)

# type (FlexPort)

Sets the FlexPort interface to support either Ethernet or Fibre Channel protocol.

## Syntax

```
type { fibre-channel | ethernet }
```

## Command Default

The default state is set to Ethernet protocol.

## Parameters

### fibre-channel

Sets the interface type to Fibre Channel protocol.

### ethernet

Sets the interface type to Ethernet protocol.

## Modes

FlexPort configuration mode

## Examples

To set the FlexPort interface type to Fibre Channel:

```
switch(config)# hardware
switch(config-hw)# flexport 1/0/1
switch(conf-hw-flex-1/0/1)# type fibre-channel
```

To set the FlexPort interface type to Ethernet:

```
switch(config)# hardware
switch(config-hw)# flexport 1/0/1
switch(conf-hw-flex-1/0/1)# type ethernet
```

## History

Release version	Command history
5.0.0	This command was introduced.

# udld enable

Enables the Unidirectional Link Detection (UDLD) protocol on an interface.

## Syntax

`udld enable`  
`no udld enable`

## Command Default

Disabled on interfaces by default.

## Modes

Interface subconfiguration mode (fo, gi, te)

## Usage Guidelines

Use **no udld enable** to unblock the interface if it has been blocked by the UDLD protocol.

## Examples

To enable UDLD on a specific tengigabitethernet interface:

```
device# configure terminal
device(config)# interface tengigabitethernet 5/0/1
device(conf-if-te-5/0/1)# udld enable
```

## Related Commands

[protocol udld](#)

# unhide built-in-self-test

Executes the built-in self test for Federal Information Processing Standards (FIPS).

## Syntax

```
unhide built-in-self-test
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Irreversible commands related to enabling FIPS compliance are hidden. Use this command to execute the built-in self test of the FIPS system.

Enter "fibranne" at the Password prompt to run the command.

This command can be entered only from a user account with the admin role assigned.

## Examples

To execute the built-in self test for FIPS:

```
switch# unhide built-in-self-test  
Password: *****
```

## History

Release version	Command history
5.0.1	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

# unhide fips

Makes available irreversible commands used in enabling Federal Information Processing Standard (FIPS) compliance.

## Syntax

```
unhide fips
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Irreversible commands related to enabling FIPS compliance are hidden. Use this command to make the following hidden commands available: **fips root disable**, **fips selftests**, **fips selftests**, and **prom-access disable**.

Enter "fibranne" at the Password prompt to run the command.

This command can be entered only from a user account with the admin role assigned.

## Examples

To make available all irreversible commands used in enabling FIPS compliance:

```
switch# unhide fips
Password: *****
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[fips selftests](#), [fips zeroize](#), [prom-access disable](#), [show fips](#)

# unlock username

Unlocks a locked user account.

## Syntax

```
unlock username name [ rbridge-id { rbridge-id | all } ]
```

## Parameters

*name*

Specifies the name of the user account.

**rbridge-id**

Specifies an RBridge or all RBridges.

*rbridge-id*

Specifies an RBridge ID.

**all**

Specifies all RBridges.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to unlock a user who has been locked out because of unsuccessful login attempts. A user account is locked by the system when the configured threshold for login retries has been reached.

## Examples

The following example unlocks a user account.

```
device# unlock username testUser  
Result: Unlocking the user account is successful
```

## Related Commands

[show running-config username](#), [show sfp](#), [username](#)

# update-time

Configures the interval at which BGP next-hop tables are modified. BGP next-hop tables should always have IGP (non-BGP) routes.

## Syntax

```
update-time sec
```

```
no update-time sec
```

## Command Default

This option is disabled.

## Parameters

sec

Update time in seconds. Range is from 0 through 30. Default is 5 seconds.

## Modes

BGP address-family IPv4 unicast configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the defaults.

The update time determines how often the device computes the routes (next-hops) in an RBridge. Lowering the value set by the **update-time** command increases the convergence rate.

By default, the device updates the BGP4 next-hop tables and affected BGP4 routes five seconds following IGP route changes. Setting the update time value to 0 permits fast BGP4 convergence for situations such as a link failure or IGP route changes, starting the BGP4 route calculation in subsecond time.

### NOTE

Use the **advertisement-interval** command to determine how often to advertise IGP routes to the BGP neighbor.

## Examples

To permit fast convergence for BGP4 globally on an RBridge (for the default VRF):

```
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv4 unicast
device(config-bgp-ipv4u)# update-time 0
```

To set the BGP4+ update-time interval to 30 for VRF instance "red":

```
device# configure terminal
device(config)# rbridge-id 10
device(config-rbridge-id-10)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# update-time 30
```

## History

Release version	Command history
5.0.0	This command was modified to add support for the IPv6 address family.
6.0.1	Support was added for the BGP address-family IPv4 and IPv6 unicast vrf configuration modes.

## Related Commands

[advertisement interval \(fabric-map\)](#)



# uplink-switch enable

Enables uplink switch protected port globally on a switch.

## Syntax

```
uplink-switch enable
no uplink-switch enable
```

## Command Default

Uplink switch protected port is disabled.

## Modes

Global configuration mode

## Usage Guidelines

Uplink switch protected port must be enabled globally before it can be enabled on an interface, by means of the **protected-port enable** command.

The **no** form of this command disables uplink protected port.

## Examples

The following example enables uplink switch protected port globally.

```
device# configure terminal
device(config)# uplink-switch enable
```

The following example disables uplink switch protected port.

```
device# configure terminal
device(config)# no uplink-switch enable
```

## History

Release version	Command history
7.2.0	This command was introduced.

# usb

Enables or disables an attached USB device. The device will be inaccessible until it is enabled.

## Syntax

```
usb { on | off }
```

## Parameters

**on**

Turns the USB device on.

**off**

Turns the USB device off.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is executed on the local device. A device reload will automatically turn the USB device off.

This command is supported only on the local device.

This command is not supported on the standby management module.

## Examples

To enable a USB device attached to the local device:

```
device# usb on
USB storage enabled
```

To disable a USB device attached to the local device:

```
device# usb off
USB storage disabled
```

## Related Commands

[usb dir](#), [usb remove](#)

# usb dir

Lists the contents of an attached USB device.

## Syntax

```
usb dir
```

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is executed on the local device. The USB device must be enabled before this function is available.

This command is supported only on the local device.

This command is not supported on the standby management module.

## Examples

To list the contents of the USB device attached to the local device:

```
switch# usb dir

firmwarekey\ 0B 2010 Aug 15 15:13
support\ 106MB 2010 Aug 24 05:36
support1034\ 105MB 2010 Aug 23 06:11
config\ 0B 2010 Aug 15 15:13
firmware\ 380MB 2010 Aug 15 15:13
Available space on usbstorage 74%
```

## Related Commands

[usb](#), [usb remove](#)

# usb remove

Removes a file from an attached USB device.

## Syntax

```
usb remove directory file file
```

## Parameters

**directory** *directory*

Specifies one the name of the directory where the file you want to remove is located. Valid USBstorage directories are /firmware, /firmwarekey, /support, and /config.

**file** *file*

Specifies the name of the file to be removed.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command is executed on the local device. The USB device must be enabled before this function is available.

This command is supported only on the local device.

This command is not supported on the standby management module.

## Examples

To remove a configuration file from a USB device attached to the local device:

```
device# usb remove directory config file startup-config.backup
```

## Related Commands

[usb](#), [usb dir](#)

# user (alias configuration)

Launches the user-level alias configuration mode, in which you can manage user aliases.

## Syntax

**user** *username*

**no user** *username*

## Parameters

*username*

Specifies the account login name.

## Modes

Alias configuration mode

## Usage Guidelines

To delete all aliases defined for a specified user, enter the **no** form of this command.

## Examples

The following example accesses user-alias configuration mode for the user `jdoe`, and defines a user-level alias named "sv" for the **show version** command.

```
device# configure terminal
device(config)# alias-config
device(config-alias-config)# user jdoe
device(config-user-jdoe)# alias sv "show version"
```

## Related Commands

[alias](#), [alias-config](#)

# username

Creates and configures a user account.

## Syntax

```
username username password password role role_name [ access-time HHMM to HHMM ] [ desc description ] [ enable { true | false } ] [ encryption-level { 0 | 7 } ] [ expire { never | YYYY-MM-DD } ]
```

```
no username name
```

## Parameters

*username*

Specifies the account login name.

**access-time** *HHMM* to *HHMM*

Restricts the hours during the day that the user may be logged in. Valid values range from 0000 through 2400. By default, users are granted 24 hour access. Use 24-hour format. For example, to restrict access to the daily work schedule, use **access-time 0800 to 1800**. By default, there is no access-time limitation. To change access time, include both the new "from" time and "to" time. To restore default access time, specify **access-time 0000 to 2400**.

**desc** *description*

Specifies a description of the account (optional). The description can be up to 64 characters long, and can include any printable ASCII character, except for the following characters: single quotation marks ('), double quotation marks ("), exclamation point (!), colon (:), and semi-colon (;). If the description contains spaces, enclose the text in double quotation marks.

**enable**

Enables or disables the account.

**true**

(Default) Enables the account.

**false**

Disables the account. A user whose account is disabled cannot log in.

**expire**

Specifies the password expiration setting.

**never**

(Default) Does not specify a password expiration date.

*YYYY-MM-DD*

Specifies a password expiration date.

**password** *password*

Specifies the account password. To use the exclamation mark (!) character, either precede it with the escape character (\)—**secret!\password**—or enclose the password within double quotes—**"secret!password"**.

**role** *role\_name*

Specifies the role assigned to the username account.

**encryption-level** { 0 | 7 }

Specifies the password encryption level. The values are 0 (clear text) and 7 (encrypted). Clear text (0) is the default. If service password-encryption is enabled, it overrides a user-level setting.

## Modes

Global configuration mode

## Usage Guidelines

The *username* must be from 1 through 40 characters. It must begin with a letter or underscore and be comprised of only letters, numbers, underscore and period. A username is case sensitive. It cannot be the same as that of an existing role.

When creating a username, you must specify a password and a role. When modifying a username, it is sufficient to enter **username** *username* , followed by the new values.

The maximum number of user accounts on a device is 64.

If a user's password, access time, or role is changed, any login sessions for that user are terminated.

To specify **access-time**, use the system time defined for the Extreme operating system. For the current system time, enter **show clock**.

To delete a user, enter the **no username** *username* command.

## Examples

The following example configures a user account.

```
device# configure terminal
device(config)# username testUser password ***** role user desc
```

The following example modifies an existing user account.

```
device# configure terminal
device(config)# username testUser desc "add op test user"
```

The following example modifies an existing user account, restricting the hours that an existing user may be logged in from 08:00 AM through 18:00 PM.

```
device# configure terminal
device(config)# username testUser access-time 0800 to 1800
```

## Related Commands

[show running-config username](#), [show users](#), [unlock username](#)

# username admin enable false

Toggles the lockout option for the default admin account.

## Syntax

**username admin enable false**

**no username admin enable false**

## Command Default

This feature is disabled.

## Modes

Global configuration mode

## Usage Guidelines

This command toggles the lockout option for the default admin account.

The account lockout policy locks an account when the user exceeds the configured number of maximum failed login attempts. This policy is now available for admin accounts. You are allowed to enable or disable lockout policy for admin accounts and user accounts with the admin role.

For admin accounts, there is no support of lockout duration to release the locked accounts. Locked admin role accounts will be reset after reboot.

Use the **no** username admin enable false command to disable this option.



# username user enable false

Toggles the lockout option for user accounts with admin privileges.

## Syntax

```
username user enable false
```

```
no username user enable false
```

## Command Default

This feature is disabled.

## Modes

Global configuration mode

## Usage Guidelines

This command toggles the enable option for user accounts with admin privileges.

The account lockout policy locks an account when the user exceeds the configured number of maximum failed login attempts. This policy is now available for admin accounts. You are allowed to enable or disable lockout policy for admin accounts and user accounts with 'admin' role.

Use the **no username user enable false** command to disable this option.

# use-v2-checksum

Enables the v2 checksum computation method for a VRRPv3 IPv4 session.

## Syntax

```
use-v2-checksum
no use-v2-checksum
```

## Command Default

VRRPv3 uses the v3 checksum computation method.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

Some non-Extreme devices only use the v2 checksum computation method in VRRPv3. This command enables v2 checksum computation method in VRRPv3 and provides interoperability with these non-Extreme devices.

The **no** form of this command enables the default v3 checksum computation method in VRRPv3 sessions.

## Examples

The following example shows the v2 checksum computation method enabled for an VRRPv3 IPv4 session on an Extreme device.

```
device(config)# rbridge-id 1
device(config-rbridge-id-1)# protocol vrrp
device(config-rbridge-id-1)# interface ve 100
device(config-Ve-100)# vrrp-group 10 version 3
device(config-vrrp-group-10)# use-v2-checksum
```

## History

Release version	Command history
5.0.1a	This command was introduced.

# vcenter

Authenticates with an established vCenter and provides additional options.

## Syntax

```
vcenter name [ activate | interval interval | { url URL username username password password } ] [ use-vrf vrf-name ]
no vcenter name [ use-vrf vrf-name ]
```

## Parameters

*name*

Name of an established vCenter.

**activate**

Activates the vCenter.

**interval**

Enables the discovery timer.

*interval*

Discovery timer interval in minutes, Range is 0 through 1440. Default is 30, and 0 disables discovery.

**url**

Enables configuration of vCenter URL, user name, and password.

*URL*

URL of the vCenter.

**username**

Configures the user name.

**password**

Configures the password.

**use-vrf** *vrf-name*

Specifies a VRF through which to communicate with the vCenter. See the Usage Guidelines.

## Modes

Global configuration mode

## Usage Guidelines

You must authenticate with an established vCenter before you can initiate any discovery transactions. In order to authenticate with a specific vCenter, you must configure the URL, login, and password properties on the VDX switch. Use this command to authenticate with a vCenter; establish a URL, username, and password; and manage discovery intervals.

Enter **no vcenter** *name* and selected operands to deactivate this functionality.

By default, all management services are enabled on the management VRF ("mgmt-vrf") and the default VRF ("default-vrf").

## Examples

To configure a vCenter and specify a VRF:

```
switch(config)# vcenter myvcenter url https://10.2.2.2 username user password pass use-vrf myvrf
switch(config)# vcenter myvcenter activate
switch(config)# no vcenter myvcenter activate
switch(config)# no vcenter myvcenter
switch(config)# vcenter myvcenter interval 60
```

## History

Release version	Command history
7.0.0	This command was modified to support the <b>use-vrf</b> keyword.

## Related Commands

[show vlan private-vlan](#)

# vcenter discovery (ignore delete responses)

Causes a vCenter to ignore delete responses.

## Syntax

```
vcenter name discovery ignore-delete-all-response [ number | always ]
```

## Command Default

The default for *number* is 0.

## Parameters

*name*

Name of the vcenter.

*number*

Number of discovery cycles to ignore. Default is 0.

**always**

Always ignore delete-all requests from the vCenter.

## Modes

Global configuration mode

## Usage Guidelines

An invalid state or condition of a vCenter can cause the deletion of all auto-port-profiles in a system. To prevent this from happening, you can configure a mode in Network OS to ignore the "delete-all" responses from the vCenter.

## Examples

```
switch(config)# vcenter vcs_demo discovery ignore-delete-all-response 3
```

# vcenter vlan-create

Manages default behavior during the vCenter discovery process, where VLANs are created automatically when they are not already present on the switch.

## Syntax

```
vcenter vlan-create { auto | switch-admin }
```

## Command Default

VLANs are created automatically during the discovery process.

## Parameters

### auto

Specifies that VLANs are created automatically during the vCenter discovery process.

### switch-admin

Specifies that the vCenter discovery process ignores port groups for which VLANs are not already established on the switch; VLANs must be established manually by the switch administrator.

## Modes

Global configuration mode

## Usage Guidelines

If a port profile is not activated, the associated VLAN must be deleted manually. Following the deletion of VLANs, you must use the **vnetwork reconcile vcenter** command on the switch and specify the name of the vCenter.

### NOTE

If a profile is activated, the deletion of VLANs in the profile is not allowed. You must delete the profile, and then delete the VLAN.

## Examples

The following example specifies that the discovery process ignores port groups for which VLANs are not already established on the switch.

```
device(config)# vcenter vlan-create switch-admin
```

The following example specifies that VLANs be created automatically during the vCenter discovery process.

## History

Release version	Command history
7.2.0	This command was introduced.

# vcs auto-config-backup timer

Initiates a backup of a running-configuration master file across all nodes in the cluster at a specified interval. The running-configuration master file is used to recover the cluster configuration in case the configuration database is lost or corrupted.

## Syntax

```
vcs auto-config-backup timer days
```

```
no vcs auto-config-backup timer
```

## Command Default

This feature is disabled.

## Parameters

*days*

Specifies the interval, in days, after which a backup of the running-configuration master file is initiated. Range is 1 through 365.

## Modes

Global configuration mode

## Usage Guidelines

The start of the interval is based on the time of configuration.

Use the **no** form of this command to remove the timer configuration and prevent an automatic backup of the running-configuration file.

You can also use the **auto-config-backup** command to initiate a manual backup of the running-configuration file at any time, in either global configuration mode or any submode. This makes it convenient to capture a critical cluster-wide configuration. The auto and manual commands overwrite the single master backup file that is maintained internally. Whichever mode initiates the most recent backup produces this file.

```
show vcs auto-config-backup
```

## Examples

To save a backup of the running-configuration automatically in global configuration mode, and specify an interval of 7 days:

```
device# configure terminal
device(config)# vcs auto-config-backup timer 7
```



To remove the timer and prevent an automatic backup in global configuration mode:

```
device# configure terminal
device(config)# no vcs auto-config-backup timer
```

**ATTENTION**

Deleting the timer also deletes the most recent master backup file.

## History

Release version	Command history
7.1.0	This command was introduced.

## Related Commands

[auto-config-backup](#), [show vcs auto-config-backup](#)

# vcs config snapshot

Takes a configuration snapshot for a specified RBridge ID.

## Syntax

```
vcs config snapshot { create | restore } rbridge-id rbridge-id snapshot-id snapshot-id
no config snapshot rbridge-id rbridge-id snapshot-id [ all | rbridge-id ]
```

## Parameters

### create

Captures the snapshot configuration from the RBridge ID specified.

### restore

Restores the snapshot configuration to the RBridge ID specified.

### rbridge-id *rbridge-id*

Specifies the RBridge ID of the configuration you are capturing in a snapshot or the RBridge ID to which you are restoring the snapshot.

### snapshot-id *snapshot-id*

Name you give to the snapshot of the configuration.

### all

Designates all of the snapshots.

## Modes

Privileged EXEC mode

## Usage Guidelines

A configuration snapshot allows you to restore the configuration if necessary. The snapshot for the RBridge specified is stored on all switches in the cluster.

The **create** and **restore** commands can be issued from any node in the cluster even though the commands pertain to a specific RBridge ID.

If a snapshot was taken on a node that had been disconnected from the cluster, the cluster will not have the snapshot. In this situation, you can use the **copy snapshot** commands to put the snapshot on the cluster.

## Examples

The following example creates a snapshot of the configuration on an RBridge with the ID of 10, and gives the name of the snapshot "snapshot10".

```
device# vcs config snapshot create rbridge-id 10 snapshot snapshot10
```

## Related Commands

[vcs virtual](#)

# vcs logical-chassis enable default-config

Use the **no** form of this command to remove a node from an existing cluster, remaining in logical chassis cluster mode.

## Syntax

```
no vcs logical-chassis enable rbridge-id rbridge-id default-config
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies the RBridge ID of the node to be removed from the cluster. Range is from 1 through 239.

**default-config**

Uses the default configuration when a node is removed from the cluster. This is a required keyword.

## Command Default

None

## Modes

Privileged EXEC mode

## Usage Guidelines

Only the **no** form of this command is supported.

This command removes the device from the logical chassis cluster and automatically changes the RBridge ID to 1 and the VCS ID to 8193, applying a default configuration. The device is removed from the original cluster, and all configurations corresponding to that node are removed from the cluster configuration database.

To return the device to the original (logical chassis) cluster, the user must set the RBridge ID and VCS ID to appropriate values for that network.

## Examples

The following example removes a device from a logical chassis cluster and applies a default configuration on RBridge 239.

```
device# no vcs logical-chassis rbridge-id 239 default-config
```

## History

Release version	Command history
7.1.0	This command was modified .

# vcs replace rbridge-id

Replaces a node in a logical chassis cluster.

## Syntax

```
vcs replace rbridge-id rbridge-id
```

## Parameters

**rbridge-id** *rbridge-id*

Specifies the RBridge ID of the node you are replacing.

## Modes

Privileged EXEC mode

## Usage Guidelines

You must re-use the RBridge ID and then enter the WWN of the new node when prompted, as shown in the example.

If you replace a node by using the **vcs replace rbridge-id** command, the "pre-provision rbridge-id <x>..." entry is added to the running configuration.

## Examples

To replace a node that has an RBridge ID of 3 and then enter the WWN of the new switch:

```
device# vcs replace rbridge-id 3
Enter the WWN of the new replacement switch: 11:22:33:44:55:66:77:81
```

# vcs set-rbridge-id

Changes the existing RBridge ID of a node, and optionally the VCS ID.

## Syntax

```
vcs set-rbridge-id rbridge-id [ vcsid vcsid ]
```

## Parameters

*rbridge-id*

Specifies a new RBridge ID.

**vcsid** *vcsid*

Specifies a new VCS ID. Range is 1 through 8192.

## Modes

Privileged EXEC mode

## Usage Guidelines

### ATTENTION

Each time you change the Extreme VCS Fabric configuration, the switch resets to the default configuration and reboots automatically. Make sure to save the configuration before you issue this command.

## Examples

To change the RBridge ID of a node to 10:

```
device# vcs set-rbridge-id 10
```

## History

Release version	Command history
6.0.1	This command was introduced.
6.0.1a	This command description was modified.
7.1.0	This command was modified to remove the " <b>logical-chassis enable</b> " keywords, to support the deprecation of fabric cluster mode.

# vcs vcsid

Changes the VCS ID of a node, and optionally the RBridge ID.

## Syntax

```
vcs vcsid vcsid [ set-rbridge-id rbridge-id ]
```

## Parameters

**vcsid** *vcsid*

Specifies a new VCS ID. Range is 1 to 8192.

**set-rbridge-id** *rbridge-id*

Specifies a new RBridge ID.

## Modes

Privileged EXEC mode

## Usage Guidelines

Each time you change the Extreme VCS Fabric configuration, the device resets to the default configuration and reboots automatically. Make sure to save the configuration before proceeding.

## Examples

To change the VCS ID of a node to 35:

```
device# vcs vcsid 35
```

To change the VCS ID of a node to 35 and specify a new RBridge ID of 10:

```
device# vcs vcsid 35 set-rbridge-id 10
```

## History

Release version	Command history
7.1.0	This command was modified to remove the " <b>logical-chassis enable</b> " and "rbridge-id" keywords, to support the deprecation of fabric cluster mode.

# vcs virtual

Assigns a single virtual IPv4 or IPv6 address to all switches in an Extreme VCS Fabric.

## Syntax

```
vcs virtual { ip address ipv4_address/prefix_len inband interface veVE_number | ipv6 address ipv6_address/prefix_len }
no vcs virtual ip address
no vcs virtual ipv6 address
```

## Parameters

**ip address** *ipv4\_address/prefix\_len*  
Specifies the IPv4 address and prefix length means of a CIDR prefix (mask).

**inband interface** *ve* *VE\_number*  
Specifies a virtual Ethernet (VE) interface.

**ipv6 address** *ipv6\_address/prefix\_len*  
Specifies the IPv6 address and prefix length means of a CIDR prefix (mask).

## Modes

Global configuration mode

## Usage Guidelines

When you configure the virtual IPv4 or IPv6 address for the first time, the address is assigned to the principal switch. You can then access the principal switch through the management port IP address or the virtual IP address. The virtual IP configuration is global in nature. All the nodes in the fabric will be configured with the same virtual IP address, but the address is always bound to the current principal switch.

This command can be used in VCS mode only after the fabric has formed successfully.

The command can be executed from any node. You can remove a virtual IP address when you are logged on to the switch through the virtual IP address. Use the management port IP address or the serial console to configure the virtual IP address.

The **inband interface** *ve* parameter can only be used when assigning an IPv4 address. This parameter is not applicable for IPv6 addresses.

It is the responsibility of the network administrator to ensure that the virtual IP address assigned is not a duplicate of an address assigned to any other management port in the VCS Fabric.

The virtual IP address should be configured on the same subnet as the management interface IP address.

Enter **no vcs virtual ip address** or **no vcs virtual ipv6 address** to remove a currently configured virtual IPv4 or IPv6 address, respectively.



## Examples

To assign a virtual IPv4 address and mask to the principal switch and specify a VE interface:

```
device(config)# vcs virtual ip address 30.30.30.14/24 inband interface ve 4
```

To remove the currently configured virtual IPv4 address:

```
device(config)# no vcs virtual ip address
```

To assign a virtual IPv6 address and mask to the principal switch

```
device(config)# vcs virtual ipv6 address 2001:db8::/64
```

## History

Release version	Command history
7.0.0	This command was modified to include support for VE interfaces and IPv6 addresses.

## Related Commands

[show vcs](#)

# vcs virtual-fabric enable

Enables the Virtual Fabrics feature, allowing the configuration of service or transport VFs in a Virtual Fabrics context. This expands the VLAN ID address space above the standard 802.1Q limit of 4095 to support multitenancy.

## Syntax

```
vcs virtual-fabric enable
```

```
no vcs virtual-fabric enable
```

## Command Default

This feature is disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

This command will be successful only if the Virtual Fabric (VF) status is VF-capable. This operation does not disrupt existing 802.1Q traffic in the fabric. Upon the successful completion of the command, the status of the fabric becomes VF-enabled.

Use the **no** form of this command to disable the configuration of service or transport VFs in the fabric. The **no** form of this command will be successful only if there is no service or transport VF configuration in the fabric and the status of the fabric is VF-enabled. All service or transport VF configurations in the fabric must be removed or the command **no vcs virtual-fabric enable** will fail. Upon successful completion of the command, the fabric status becomes VF-capable.

On the Extreme VDX family of switches, VLANs are treated as interfaces from a configuration point of view. By default, all the DCB ports are assigned to VLAN 1 (VLAN ID equals 1). Valid VLAN IDs are as follows:

- On Extreme VDX 8770 switches: 1 through 4086 for 802.1Q VLANs (VLAN IDs 4087 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.
- On all other Extreme VDX switches: 1 through 3962 for 802.1Q VLANs (VLAN IDs 3963 through 4095 are reserved on these switches), and 4096 through 8191 for service or transport VFs in a Virtual Fabrics context.

### NOTE

When the fabric is VF-enabled with a VF-specific configuration, the user is advised of relevant errors when attempting to disable the VF.

## Examples

To enable the Virtual Fabrics feature:

```
device(config)# vcs virtual-fabric enable
```

To disable the Virtual Fabrics feature when there is no classified VLAN configuration in the fabric:

```
device(config)# no vcs virtual-fabric enable
```

## Related Commands

[interface vlan](#)

## version

Specifies the OpenFlow version to be used.

## Syntax

```
version ofv130
```

## Modes

OpenFlow logical-instance configuration mode

## Usage Guidelines

OpenFlow v1.3 is the only version currently available. This is also the default, so this does not need to be configured.

## Examples

The following example enters OpenFlow logical-instance configuration mode and specifies the available OpenFlow version.

```
device(config)# rbridge-id 12
device(config-rbridge-id-12)# openflow logical-instance 1
device(config-logical-instance-1)# version ofv130
device(config-logical-instance-1)#
```

## History

Release version	Command history
6.0.1	This command was introduced.

# virtual-fabric

Designates the FCoE Virtual Fabric Identification (VFID).

## Syntax

```
virtual-fabric 1
```

## Modes

FCoE fabric-map configuration mode

## Usage Guidelines

The VFID value is 1. No other values are allowed.

You must be in the feature configuration mode for FCoE fabric-map for this command to function.

### NOTE

The FCoE virtual fabric is not to be confused with the Virtual Fabric feature that supports service or transport VFs.

## Examples

```
switch(config)# fcoe
switch(config-fcoe)# fabric-map default
switch(config-fcoe-fabric-map)# virtual-fabric 1
```

# virtual-ip

Configures a virtual IPv4 address or IPv6 address for the virtual router.

## Syntax

```
virtual-ip { ipv4-address | ipv6-address }
```

```
no virtual-ip { ipv4-address | ipv6-address }
```

## Parameters

*ipv4-address*

Virtual IPv4 address of the virtual router.

*ipv6-address*

Virtual IPv6 address of the virtual router.

## Modes

Virtual-router-group configuration mode

## Usage Guidelines

The virtual IPv4 address or IPv6 address is the IP address that an end-host sets as its default gateway. The virtual IP address must belong to the same subnet as the underlying interface. A maximum of 16 virtual IP addresses can be configured for VRRP; only one virtual IP address can be configured for VRRP-E. The session is enabled as soon as the first virtual IP address is configured.

You can perform this command for VRRP or VRRP-E. VRRPv3 introduced the ability to use an IPv6 address when an IPv6 VRRPv3 group is configured.

This command accepts both fe80/10 link local addresses or fe80/64 addresses as virtual-IP.

Enter the **no virtual-ip** command with a specified virtual IP address to delete the specified virtual IP address

## Examples

To assign a virtual IP address of 192.53.5.1 to the VRRP virtual group 1:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp
device(config-rbridge-id-101)# interface te 101/1/6
device(config-if-te-101/1/6)# vrrp-group 1
device(config-vrrp-group-1)# virtual-ip 192.53.5.1
```

To assign a virtual IP address of 192.53.5.1 to the VRRP-E virtual group 1:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp
device(config-rbridge-id-101)# int ve 20
device(config-ve-20)# vrrp-group-extended 1
device(config-vrrp-extended-group-1)# virtual-ip 192.53.5.1
```

To assign a virtual IPv6 address of 2001:2019:8192::1 to the VRRP-Ev3 virtual group 19:

```
device(config)# rbridge-id 122
device(config-rbridge-id-122)# ipv6 protocol vrrp-extended
device(config-rbridge-id-122)# interface ve 2019
device(config-ve-2019)# ipv6 address 2001:2019:8192::122/64
device(config-ve-2019)# ipv6 vrrp-extended-group 19
device(config-vrrp-extended-group-19)# virtual-ip 2001:2019:8192::1
```

## Related Commands

[vrrp-group](#)

# virtual-mac

Enables generation of a virtual MAC with 0 IP hash.

## Syntax

**virtual-mac** *virtual\_mac\_address*

## Parameters

*virtual\_mac\_address*

Specifies a virtual MAC address.

## Modes

VRRP-Extended group configuration mode

## Usage Guidelines

The distributed VXLAN gateway functionality depends on VRRP-E for multi-homing. By default, the VRRP-E virtual MAC is derived as 02:e0:52:<2-byte-ip-hash>:<1-byte-vid>. The VXLAN gateway requires that the virtual MAC be a function of only VRID. The two-byte hash of the virtual IP should be set to zeros, for example, 02e0.5200.00xx:100.

## Examples

To enable the generation of a virtual MAC with 0 IP hash:

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# int ve 10
device(config-Ve-10)# vrrp-extended-group 100
device(config-vrrp-extended-group-100)# virtual-mac 02e0.5200.00xx:100
```



# vlag ignore-split

Controls the ignore-split recovery functionality.

## Syntax

**vlag ignore-split**

**no vlag ignore-split**

## Command Default

**vlag ignore-split** is enabled.

## Modes

Port-channel configuration mode

## Usage Guidelines

When ignore-split-recovery is active, neither of the RBridges modify their actor SID when splitting or rejoining the vLAG. They both advertise VSID and keep both sides of the vLAG alive.

Enter **no vlag ignore-split** to disable this functionality.

### NOTE

It is recommended that this command be enabled.

## Examples

```
device(config)# interface port-channel 27
device(config-port-channel-27)# vlag ignore-split
```

# vlag-commit-mode disable

Disables the virtual LAG (vLAG) commit mode for dynamic vLAGs, achieving scalability to support a resilient network infrastructure. The command also disables the actor and partner SID selection operations.

## Syntax

```
vlag-commit-mode disable
```

```
no vlag-commit-mode disable
```

## Command Default

The vLAG commit mode is enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The vLAG commit mode cannot be disabled if an RBridge in the Extreme VCS Fabric is running a version earlier than Network OS 7.0.0.

### NOTE

The **no vlag ignore-split** command cannot be executed when the **vlag-commit-mode disable** command is used. Alternately, the **vlag-commit-mode disable** command cannot be executed if the **no vlag ignore-split** command is in effect.

**no**

Use the **show running-config vlag-commit-mode disable** command to confirm the results of this command.

## Examples

The following example disables the vLAG commit mode on an Extreme device and enables scalability.

```
device# configure terminal
device(config)# vlag-commit-mode disable
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[show running-config vlag-commit-mode](#)

# vlan classifier activate group

Activates a VLAN classifier group.

## Syntax

```
vlan classifier activate group number vlan vlan_id  
no vlan classifier activate group number
```

## Parameters

*number*

Specifies which VLAN classifier group to activate. Valid values range from 1 through 16.

vlan *vlan\_id*

Specifies which VLAN interface to activate.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no vlan classifier activate group *number*** to remove the specified group.

## Examples

To activate VLAN classifier group 1 for VLAN 5 on a specific 10-gigabit Ethernet interface:

```
switch(config)# interface tengigabitethernet 178/0/1  
switch(conf-if-te-178/0/1)# vlan classifier activate group 1 vlan 5
```

To remove VLAN classifier group 10 from a specific port-channel interface:

```
switch(config)# interface port-channel 44  
switch(config-port-channel-44)# no vlan classifier activate group 10
```

## Related Commands

[interface](#), [vlan classifier group](#)

# vlan classifier group

Adds and deletes rules to a VLAN classifier group.

## Syntax

```
vlan classifier group number [ add rule number | delete rule number ]
```

## Parameters

*number*

Specifies the VLAN group number for which rules are to be added or deleted. Valid values range from 1 through 16.

**add rule** *number*

Specifies a rule is to be added. Valid values range from 1 through 256.

**delete rule** *number*

Specifies a rule is to be deleted. Valid values range from 1 through 256.

## Modes

Global configuration mode

## Usage Guidelines

Make sure your converged mode interface is not configured to classify untagged packets to the same VLAN as the incoming VLAN-tagged packets. By configuring a converged interface to classify untagged packets (by using classifiers or the default port *vlan\_id*) to the same VLAN as VLAN-tagged packets coming into the interface, the FCoE hardware sends out untagged packets to the CNA. These packets may be dropped, disrupting communications.

## Examples

To add rule 1 to VLAN classifier group 1:

```
switch(config)# vlan classifier group 1 add rule 1
```

# vlan classifier rule

Creates a VLAN classifier rule to dynamically classify Ethernet packets on an untagged interface into VLANs.

## Syntax

```
vlan classifier rule rule_id [[ mac mac_address ] | [ proto { hex_addr encap { ethv2 | nosnapllc | snapllc } | arp encap { ethv2 | nosnapllc | snapllc } | ip encap { ethv2 | nosnapllc | snapllc } | ipv6 encap { ethv2 | nosnapllc | snapllc } ] ]
```

```
no vlan classifier rule
```

## Parameters

*rule\_id*

Specifies the VLAN identification rule. Valid values range from 1 through 2556.

**mac**

Specifies the Media Access Control (MAC) list.

*mac\_address*

Specifies the MAC address-based VLAN classifier rule used to map to a specific VLAN.

**proto**

Specifies the protocol to use for the VLAN classifier rule.

*hex\_addr*

An Ethernet hexadecimal value. Valid values range from 0x0000 through 0xffff

**arp**

Specifies to use the Address Resolution Protocol.

**ip**

Specifies to use the Internet Protocol.

**ipv6**

Specifies to use the Internet Protocol version 6.

**encap**

Specifies to encapsulate the Ethernet frames sent for the VLAN classifier rule.

**ethv2**

Specifies to use the Ethernet version 2 encapsulated frames.

**nosnapllc**

Specifies to use the Ethernet version 2 non-SNA frames.

**snapllc**

Specifies to use the Ethernet version 2 with SNA frames.

## Modes

Global configuration mode

## Usage Guidelines

VLAN classifiers are created individually and are managed separately. Up to 256 VLAN classifiers can be provisioned. One or more VLAN classifiers can be grouped into a classifier group. This classifier group can further be applied on an interface.

Enter **no vlan classifier rule** *rule\_id* to delete the specified rule.

## Examples

To create an ARP VLAN classifier rule:

```
switch(config)# vlan classifier rule 2 proto arp encap ethv2
```

## Related Commands

[show vlan](#)

# vlan dot1q tag native

Enables 802.1Q tagging on the native VLAN on all trunked ports on the switch.

## Syntax

```
vlan dot1q tag native
```

```
no vlan dot1q tag native
```

## Command Default

The native VLAN is enabled.

## Modes

Global configuration mode

## Usage Guidelines

Usually, you configure 802.1Q trunks with a native VLAN ID, which strips tagging from all packets on that VLAN.

To maintain the tagging on the native VLAN and drop untagged traffic, use the **vlan dot1q tag native** command. The switch will tag the traffic received on the native VLAN and admit only 802.1Q-tagged frames.

Control traffic continues to be accepted as untagged on the native VLAN on a trunked port, even when the **vlan dot1q tag native** command is enabled.

Enter **no vlan dot1q tag native** to disable dot1q (IEEE 802.1Q) tagging for all native VLANs on all trunked ports on the switch.

## Related Commands

[switchport mode](#), [switchport trunk allowed vlan rspan-vlan](#)

# vlan-profile (AMPP)

Activates the VLAN profile mode for AMPP.

## Syntax

```
vlan-profile
no vlan-profile
```

## Modes

Port-profile configuration mode

## Usage Guidelines

The VLAN profile mode for AMPP allows configuration of VLAN attributes of a port-profile.

Enter **no vlan-profile** to delete the profile.

## Examples

To create a basic VLAN profile supporting 802.1Q VLANs:

```
switch(config)# port-profile my_profile
switch(conf-port-profile-my_profile)# vlan-profile
```

The following illustrates the creation of port-profiles and vlan-profiles with switchport configurations illustrating VLAN classifications in a Virtual Fabrics context:

```
switch(config)# port-profile pp100
switch(config-port-profile-pp100)# vlan-profile
switch(config-vlan-profile)# switchport
switch(config-vlan-profile)# switchport access vlan 5001 mac 1.1.1
switch(config-vlan-profile)# switchport access vlan 5002 mac 1.1.2
switch(config-vlan-profile)# switchport access vlan 5001 mac-group 11
switch(config-vlan-profile)# port-profile pp101
switch(config-port-profile-pp101)# vlan-profile
switch(config-vlan-profile)# switchport
switch(config-vlan-profile)# switchport mode trunk
switch(config-vlan-profile)# switchport trunk allowed vlan add 5004 ctag 11
switch(config-vlan-profile)# switchport trunk allowed vlan add 5005 ctag 12
switch(config-vlan-profile)# switchport trunk native-vlan 5006 ctag 13
```



# vnetwork reconcile vcenter

Synchronizes the device configuration with the configuration discovered by vCenter.

## Syntax

```
vnetwork reconcile vcenter vCenter_name
```

## Command Default

The vNetwork information is reconciled.

## Parameters

```
vcentervcenter_name  
Specifies a vCenter.
```

## Modes

Privileged EXEC mode

## Usage Guidelines

Run the **show vnetwork diff vcenter** command to discover if there are any configuration discrepancies between the current device and the data center. If there are, run the **vnetwork reconcile vcenter** command to correct the discrepancies.

## Examples

This example shows that the profiles are not listed in the running-config.

```
device# show vnetwork diff vcenter wpgdc1vcenter
port-profiles not created on Switch
-----
auto_wpgdc1vcenter_datacenter-2_128_Network
auto_wpgdc1vcenter_datacenter-2_192_Network-vlan660
auto_wpgdc1vcenter_datacenter-2_64_Network-vlan661

device# vnetwork reconcile vcenter wpgdc1vcenter
!
device#
```

## History

Release version	Command history
5.0.2b	This command was introduced.

# vnetwork vcenter discover

Explicitly starts the discovery process on the vCenter.

## Syntax

```
vnetwork vcenter vcenter_name discover
```

## Parameters

*vcenter\_name*

Name of a vCenter.

## Modes

Privileged EXEC mode

## Usage Guidelines

The discovery of virtual assets from the vCenter occurs during one of the following circumstances:

- When a switch boots up.
- When a new vCenter is configured on the VDX switch and activated (activation turns on the timer processing, set to 180-second intervals.)

When the discovery is explicitly initiated with the CLI.

# vni (EVPN)

Enters Virtual Network Identifier (VNI) configuration mode.

## Syntax

**vni** *number*

**no vni** *number*

## Parameters

*number*

Specifies a VNI and enters VNI configuration mode.

## Modes

EVPN instance configuration mode

## Usage Guidelines

Use the **vni** *number* command once the VNI has been added to the EVPN instance using **vni add** *vni-range* command to enter VNI configuration mode where a route-distinguisher (RD) and route target (RT) can be configured for the EVPN instance and routes can be imported or exported. Refer to the **route-target (VNI)** and the **rd (VNI)** commands for more information.

The **no** form of the command removes the configuration for a particular VNI under an EVPN instance.

## Examples

The following example adds a VNI for the EVPN instance "myinstance" and enters VNI configuration mode .

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# vni add 100
device(config-evpn-instance-myinstance)# vni 100
device(evpn-vni-100)#
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[evpn-instance](#)

## vni (VRF)

Adds and removes Layer 3 virtual network identifiers (VNIs) for a VRF instance.

### Syntax

**vni** *number*

**no vni** *number*

### Command Default

Disabled

### Parameters

*number*

Specifies a VNI. Valid values range from 1 through 16777215.

### Modes

VRF configuration mode

### Usage Guidelines

A VNI number must be unique across all VRF instances within a node. The **no** form of the command removes a VNI from the VRF instance.

### Examples

The following example assigns a VNI of 100 for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf red
device(config-vrf-red)# vni 100
```

The following example removes the configured VNI for VRF instance "red".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf red
device(config-vrf-red)# no vni 100
```

### History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[vrf](#)

## vni add

Adds Virtual Network Identifiers (VNIs) for an EVPN instance.

### Syntax

**vni add** *vni-range*

**no vni add** *vni-range*

### Parameters

*vni-range*

Adds a range of VNIs to this EVPN instance. Valid values range from 1 through 16777215.

### Modes

EVPN instance configuration mode

### Usage Guidelines

Use the **vni add** *vni-range* command to add VNIs for an EVPN instance. Once VNIs have been added to the EVPN instance, the **vni number** command can be used to enter VNI configuration mode where a route-distinguisher (RD) and route target (RT) can be manually configured for the EVPN instance and routes can be imported or exported.

The **no** form of the command removes a VNI from the EVPN instance.

#### NOTE

Ensure that only local VNIs are added, supporting local VLANs. With symmetric integrated routing and bridging (IRB), if a remote VNI is added that is not on the local switch, that VNI is treated as being used for Layer 2 extension and ARP entries are not programmed in hardware, resulting in potential performance problems. For example, if switch Switch-1 has only VNI 100 and another switch, Switch-2, has VNI 200, you would execute the command **vni add 100** on Switch-1 and the command **vni add 200** on Switch-2.

### Examples

The following example adds a VNI for the EVPN instance “myinstance” and enters VNI configuration mode .

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# vni add 100
device(config-evpn-instance-myinstance)# vni 100
device (evpn-vni-100) #
```

### History

Release version	Command history
7.0.0	This command was introduced.

Release version	Command history
7.1.0	The Usage Guidelines were modified to clarify that only local VNIs must be added.

## vni remove

Removes Virtual Network Identifiers (VNIs) for an EVPN instance.

### Syntax

**vni remove** *vni-range*

**no vni remove** *vni-range*

### Parameters

**remove** *vlan-range*

Removes a range of VNIs from this EVPN instance. Valid values range from 1 through 16777215.

### Modes

EVPN instance configuration mode

### Usage Guidelines

Use the **remove** *vni-range* parameters to remove VNIs for an EVPN instance.

### Examples

The following example removes a range of VLANs for the EVPN instance "myinstance".

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# evpn-instance myinstance
device(config-evpn-instance-myinstance)# vni remove 1-30
```

### History

Release version	Command history
7.0.0	This command was introduced.



# vrf

Creates a Virtual Routing and Forwarding (VRF) instance and enters VRF configuration mode.

## Syntax

**vrf** *name*

## Parameters

*name*

Character string for the name of the VRF. The string can be up to 24 characters long, but should not contain punctuation or special characters.

## Modes

RBridge ID configuration mode

## Examples

To create the VRF instance "myvrf" and enter VRF configuration mode:

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# vrf myvrf
device(config-vrf-myvrf)#
```

## History

Release version	Command history
7.0.1	This command was modified to include an example.

# vrf forwarding

Configures any port as a VRF port.

## Syntax

```
vrf forwarding vrf_name
```

```
no vrf forwarding vrf_name
```

## Parameters

*vrf\_name*

The name of the VRF option for the port.

## Command Default

By default, the out-of-band (OOB) management port (the eth0 interface) is part of the pre-defined VRF named mgmt-vrf.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The "no" form of this command disables this VRF.

## Examples

To enable the management VRF on an Ethernet interface and assign the interface to a subnet:

```
switch(config)# int te 3/0/2
switch(conf-if-te-3/0/2)# vrf forwarding mgmt-vrf
switch(conf-if-te-3/0/2)# ip addr 10.1.1.1/24
```

To disable a management VRF previously configured on a VE interface:

```
switch(config)# int ve 100
switch(conf-Ve-100)# no vrf forwarding mgmt-vrf
```

## History

Release version	Command history
5.0.0	This command was introduced.

# vrf-lite-capability

Disables the down bit (DN bit) that is set when routes are redistributed from multiprotocol BGP (MP-BGP) to OSPF.

## Syntax

```
vrf-lite-capability
no vrf-lite-capability
```

## Modes

OSPF router configuration mode  
OSPF router VRF configuration mode

## Usage Guidelines

A Customer Edge (CE) router acts as the Provider Edge (PE) router in VRF Lite. Because PE routers advertise VPN routes to CE routers with the DN-bit set, these checks should be disabled in a VRF Lite context.

When a type 3, 5, or 7 link-state advertisement (LSA) is sent from a PE router running multiprotocol BGP to a CE router, the DN (down) bit in the LSA options field must be set. This prevents any type 3, 5, or 7 LSA messages sent from the CE router to the PE router from being distributed any farther. The PE router ignores messages with the DN bit set and does not add these routes to the VRF (Virtual Routing and Forwarding) routing table.

### NOTE

The **vrf-lite-capability** command is only present in OSPF router configuration mode and OSPF router VRF configuration mode. Therefore, when a BGP route is redistributed from an MPLS domain into OSPFv3 and the DN bit is set, the routes must be installed in the OSPFv3 routing table. Such routes could get propagated back into the MPLS cloud if there are OSPFv3 back-door links configured.

Enter **no vrf-lite-capability** to disable this feature.

## Examples

```
device# configure terminal
device(config)# rbridge-id 5
device(config-rbridge-id-5)# router ospf vrf orange
device(config-router-ospf-vrf-orange)# vrf-lite-capability
```

# vrf mgmt-vrf

Configures routes on a management VRF port.

## Syntax

vrf mgmt-vrf

## Command Default

None

## Modes

RBridge ID configuration mode

## Usage Guidelines

The management VRF is a dedicated, secure VRF instance that allows users to manage the router inband on switched virtual interfaces (SVIs) and physical interfaces. The name of this VRF instance is "mgmt-vrf;" this instance cannot be deleted.

A management port is any port that is part of the management VRF. The OOB port cannot be removed from the management VRF. In addition, Layer 3 virtual and physical ports (also known as front-end or inband ports) can be part of the management VRF. Inband ports can be moved, by means of the CLI, into and out of the management VRF.

## Examples

The following configures an IPv4 route subnet for the management VRF, enters address family IPv4 configuration mode, and assigns the management VRF to an Ethernet interface.

```
switch(config)# rbridge-id 3
switch(config-rbridge-id-3)# vrf mgmt-vrf
switch(config-vrf-mgmt-vrf)# ip route 10.1.1.0/32 te 3/0/10
```

## History

Release version	Command history
5.0.0	This command was introduced.

# vrrp-acceptmode-disable

Disables accept mode for the backup Virtual Router Redundancy Protocol (VRRP) virtual IP (VIP).

## Syntax

```
vrrp-acceptmode-disable  
no vrrp-acceptmode-disable
```

## Command Default

When configured, accept mode is enabled by default.

## Modes

RBridge configuration mode

## Usage Guidelines

Accept mode—enabled by default—allows a backup VRRP master device to respond to ping, traceroute, and Telnet packets if it becomes the master VRRP device. This command disables accept mode.

The **no** form of the command re-enables accept mode for the backup VRRP VIP.

## Examples

The following example shows how to disable accept mode for the backup VRRP VIP.

```
device# configure terminal  
device (config)# rbridge-id 1  
device(config-rbridge-id-1)# vrrp-acceptmode-disable
```

The following example shows how to re-enable accept mode for the backup VRRP VIP.

```
device# configure terminal  
device (config)# rbridge-id 1  
device(config-rbridge-id-1)# no vrrp-acceptmode-disable
```

# vrrp-extended-group

Configures a virtual-router-extended group and enters into the virtual router configuration mode..

## Syntax

```
vrrp-extended-group group-ID
```

```
no vrrp-extended-group group-ID
```

## Parameters

*group-ID*

A user-assigned number from 1 through 255 that you assign to the virtual router group.

## Modes

Virtual Ethernet (ve) interface configuration mode

## Usage Guidelines

This configuration is for virtual Ethernet (ve) interfaces only.

Enter **no vrrp-extended-group** *group-ID* to remove the specific VRRP Extended group.

If you remove a group, you cannot retrieve it. You would have to redo the configuration procedure.

## Examples

The following example shows how to assign the ve interface with a vlan number of 20 to the virtual router extended group with the ID of 1. (First you must enable VRRP-E on the switch.)

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp-extended
device(config-rbridge-id-101)# int ve 20
device(config-ve-20)# vrrp-extended-group 1
```

## Related Commands

[interface](#), [interface ve](#), [virtual-ip](#)

# vrrp-group

Configures a virtual router group (VRRP) and enters into the virtual router configuration mode.

## Syntax

```
vrrp-group group-ID [ version { 2 | 3 } ]
```

```
no vrrp-group group-ID [ version { 2 | 3 } ]
```

## Command Default

VRRP version 2 is the default.

## Parameters

*group-ID*

A value from 1 through 255 that you assign to the virtual router group.

**version**

Specifies in which version of VRRP the IPv4 VRRP group is to be configured.

2 | 3

Version 2 or version 3 of VRRP.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Enter **no vrrp-group** *group-ID* to remove a specific VRRP group. If you remove a group, you cannot retrieve it. You would have to redo the configuration procedure.

You can specify in which version of VRRP the VRRP group is configured using the **version** keyword and either 2 or 3 as the version number. VRRPv3 supports both IPv4 and IPv6 addresses.

## Examples

The following example shows how to assign an Ethernet interface to the virtual router group with the ID of 1. (First you must enable VRRP on the switch.)

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp
device(config-rbridge-id-101)# interface tengigabitethernet 101/1/6
device(config-if-te-101/1/6)# vrrp-group 1
```

The following example shows how to assign an Ethernet interface to the virtual router group with the ID of 1 for VRRPv3. (First you must enable VRRP on the switch.)

```
device(config)# rbridge-id 101
device(config-rbridge-id-101)# protocol vrrp
device(config-rbridge-id-101)# interface te 101/1/6
device(conf-if-te-101/1/6)# vrrp-group 1 version 3
```

## Related Commands

[interface](#)



# vtep-discovery

Enables automatic VXLAN tunnel endpoint (VTEP) discovery by BGP.

## Syntax

**vtep-discovery**

**no vtep-discovery**

## Command Default

Enabled.

## Modes

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

The **no** form of this command disables automatic VTEP discovery and creation of VXLAN tunnels.

## Examples

The following example disables automatic VTEP discovery by BGP.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# no vtep-discovery
```

The following example re-enables automatic VTEP discovery and creation of VXLAN tunnels by BGP.

```
device# configure terminal
device(config)# rbridge-id 1
device(config-rbridge-id-1)# router bgp
device(config-bgp-router)# address-family l2vpn evpn
device(config-bgp-evpn)# vtep-discovery
```

## History

Release version	Command history
7.0.0	This command was introduced.

## Related Commands

[address-family l2vpn evpn](#), [router bgp](#)

# write erase

Returns switch to factory default state.

## Syntax

```
write erase [ rbridge-id rbridge-ID ] [ vcs-id vcs_id ] [ vcs-mode { 1 } ]
```

## Command Default

None

## Parameters

**rbridge-id** *rbridge-id*

After reboot, the switch is set to the specified RBridge ID

**vcs-id** *vcs\_id*

After reboot, the switch is set to the specified VCS ID.

**vcs-mode**

Specifies the mode for the switch after it reboots.

**1**

The switch comes back up in logical chassis cluster mode (the only available mode).

## Modes

Privileged EXEC mode

## Usage Guidelines

The command can be run only on the active MM. Both MMs will be brought to the specified factory default state, with the following guidelines:

- If you do not specify any optional parameters, this command resets all user configurations, including the management IP address and all licenses.
- If you specify any optional parameters, this command resets all user configurations except the management IP address and licenses and the updates to use the parameters that you specified.

This command can be used for switch recovery or switch configuration reset to the factory default state. Due to its disruptive nature, this command prompts the user about the consequence of losing all current user configuration and resetting the switch to the factory default state. It waits for the user's confirmation before proceeding.

## Examples

To reset the switch to factory defaults, and to bring the switch back up in logical chassis cluster mode, and with an RBridge ID of 25:

```
switch# write erase rbridgeid 25 vcs-mode 1
```

## History

Release version	Command history
5.0.0	This command was introduced.
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[firmware download](#), [preprovision](#), [show bare-metal](#)

# zoning defined-configuration alias

Creates or deletes a zone alias, adds one or more members to a zone alias, or removes a member from a zone alias.

## Syntax

**zoning defined-configuration alias** *aliasName*

**member-entry** *member* [ ; *member* ] ...

**no member-entry** *member*

**no zoning defined-configuration alias** *aliasName*

## Parameters

**alias** *aliasName*

Specifies a zone alias.

**member-entry** *member*

Specifies the WWN of the device to be added to the zone alias.

## Modes

Global configuration mode

Zoning configuration mode

## Usage Guidelines

This command enters a subconfiguration mode, where you can specify the names of the zone alias members to be added to the defined configuration or removed from the defined configuration.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric. Edits can be performed only from the principal R Bridge.

Enter the **member-entry** command with one or more specified members in the subconfiguration mode to add one or more members to a zone alias. You specify the zone alias member by its WWN. When adding multiple members in a single command line, the members are added sequentially until all members are added or the first error is encountered.

Enter **no member-entry** *member* to remove a member from a zone alias. You can remove only one member entry each time you enter **no member-entry**.

If you remove the last member from a zone alias and subsequently commit the configuration, the commit operation deletes the zone alias.

Enter **no zoning defined-configuration alias** *aliasName* to delete a zone alias.

The **zoning defined-configuration alias** command changes the defined configuration.

To save the configuration persistently, enter **zoning enabled-configuration cfg-action cfg-save** . For the change to become effective, enable the configuration by entering **zoning enabled-configuration cfgName**.

## Examples

To create a zone alias with one member:

```
device(config)# zoning defined-configuration alias alias1
device(config-alias-alias1)# member-entry 10:00:00:00:00:00:01
```

To add two additional WWNs to the zone alias:

```
device(config)# zoning defined-configuration alias alias1
device(config-alias-alias1)# member-entry 10:00:00:00:00:00:02;10:00:00:00:00:00:03
```

To remove a WWN from a zone alias:

```
device(config)# zoning defined-configuration alias alias1
device(config-alias-alias1)# no member-entry 10:00:00:00:00:00:01
```

To delete a zone alias from the defined configuration:

```
device(config)# no zoning defined-configuration alias alias1
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[show running-config zoning defined-configuration](#)

# zoning defined-configuration cfg

Creates a new zone configuration or adds a zone to an existing configuration.

## Syntax

```
zoning defined-configuration cfg cfgName
member-zone zoneName [ ;zoneName ] ...
no member-zone zoneName
no zoning defined-configuration cfg cfgName
```

## Parameters

*cfgName*

Specifies the name of the zone configuration.

**member-zone** *zoneName*

Specifies the name of a zone to be added to the configuration or removed from the configuration.

## Modes

Global configuration mode

Zoning configuration mode

## Usage Guidelines

This command enters a subconfiguration mode where you can specify the names of the member zones to be added to the configuration or removed from the configuration.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric. Edits can be performed only from the principal R Bridge.

The **zoning defined-configuration cfg** command changes the defined configuration. To save the configuration persistently, issue the **zoning enabled-configuration cfg-action cfg-save** command. For the change to become effective, enable the configuration with the **zoning enabled-configuration cfg-name** command.

Enter the **member-zone** command with one or more zone names in the subconfiguration mode to add additional zones. When adding multiple zones in a single command line, the zones are added sequentially until all zones are added or the first error is encountered.

Enter the **no member-zone** with a specified zone name in the subconfiguration mode to remove a member zone from the configuration. You can remove only one member zone each time you enter the **no member-zone** command.

If you enable a zone configuration, the members in that zone configuration must be populated with at least one zone member-entry (a WWN or an alias); otherwise the enable operation fails.

If you remove the last zone from the configuration and subsequently commit the configuration, the commit operation deletes the configuration.

Enter **no zoning defined-configuration cfg *cfgName*** to delete a configuration.

#### NOTE

Zone aliases are not valid zone configuration members. Adding an alias to an existing zone configuration will not be blocked. However, the attempt to enable a zone configuration that contains aliases will fail with an appropriate error message.

## Examples

To add two zones to a zone configuration:

```
device(config)# zoning defined-configuration cfg cfg4
device(config-cfg-cfg4)# member-zone zone2;zone3
```

To delete a zone from a zone configuration:

```
device(config)# zoning defined-configuration cfg cfg4
device(config-cfg-cfg4)# no member-zone zone2
```

To delete a zone configuration:

```
device(config)# no zoning defined-configuration cfg cfg4
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[zoning defined-configuration zone](#), [zoning enabled-configuration \*cfg-name\*](#)

# zoning defined-configuration zone

Creates a new zone or adds a member to an existing zone.

## Syntax

**zoning defined-configuration zone** *zoneName*

**member-entry** *member* [ ; *member* ] ...

**no member-entry** *member*

**no zoning defined-configuration zone** *zoneName*

## Parameters

**zone** *zoneName*

Specifies the name of the zone to be configured.

**member-entry** *member*

Specifies the name of the zone member to be added to the zone. The zone member can be specified by a WWN or a by a zone alias.

## Modes

Global configuration mode

Zoning configuration mode

## Usage Guidelines

Use this command to create a new zone, to add one or more members to a zone, or to delete a member from a zone.

This command enters a subconfiguration mode, where you can specify the names of the zone members to be added to the defined configuration or removed from the defined configuration.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric. Edits can be performed only from the principal R Bridge.

You can define a zone member by its port WWN or node WWN. However, when creating an LSAN zone, you should use only port WWNs, because node WWNs cannot be exported to a remote fabric.

Enter the **member-entry** command with one or more zone member names in the subconfiguration mode to add additional members to a zone. You can specify the zone member by its WWN or by a zone alias. When adding multiple members in a single command line, the members are added sequentially until all members are added or the first error is encountered.

Enter **no member-entry** *member* to remove a member from a zone. You can remove only one member entry each time you enter **no member-entry**.

If you enable a zone configuration, the members in that zone configuration must be populated with at least one member-entry; otherwise the enable operation fails. However, You can have a non-populated zone in a zone configuration if that zone configuration only exists in the defined-configuration and is not enabled.



If you remove the last member from a zone and subsequently commit the configuration, the commit operation deletes the zone.

Enter **no zoning defined-configuration zone zoneName** to delete a zone.

The **zoning defined-configuration zone** command changes the defined configuration:

- To save the configuration persistently, issue the **zoning enabled-configuration cfg-action cfg-save** command.



#### CAUTION

For the change to become effective, enable the configuration with the **zoning enabled-configuration cfg-name** command. When edits are made to the defined configuration, and those edits affect a currently enabled zone configuration, issuing a "cfg-save" command makes the enabled configuration effectively stale. Until the enabled configuration is re-enabled, the merging of new RBridges into the cluster is not recommended. This merging may cause unpredictable results, with the potential for mismatched enabled-zoning configurations among the RBridges in the cluster.

## Examples

To add a WWN and an alias to a zone:

```
device(config)# zoning defined-configuration zone zone3
device(config-zone-zone3)# member-entry 11:22:33:44:55:66:77:84;alias1
```

To remove a WWN from a zone:

```
device(config)# zoning defined-configuration zone zone3
device(config-zone-zone3)# no member-entry 11:22:33:44:55:66:77:82
```

To remove an alias from a zone:

```
device(config)# zoning defined-configuration zone zone3
device(config-zone-zone3)# no member-entry alias1
```

To delete a zone from the defined configuration:

```
device(config)# no zoning defined-configuration zone zone3
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[zoning defined-configuration cfg](#)

# zoning enabled-configuration cfg-action cfg-clear

Clears all zone configurations in the defined configuration.

## Syntax

```
zoning enabled-configuration cfg-action cfg-clear
```

## Modes

Global configuration mode

## Usage Guidelines

The enabled configuration is not affected by this command.

Zoning configuration data are automatically distributed among all RBridges in the VCS Fabric.

This command can be entered on any RBridge in an Extreme VCS Fabric. Edits can be performed only from the principal RBridge.

After clearing the defined zone configuration with **zoning enabled-configuration cfg-action cfg-clear**, enter **no zoning enabled-configuration cfg-name** to clear the entire zoning configuration (both the defined zone configuration and the enabled configuration).

If no current active zoning configuration exists, or you just want to clear the defined zone configuration, enter **zoning enabled-configuration cfg-action cfg-save** to commit the transaction.

If a **cfg-clear** is entered accidentally, issue a **cfg-transaction-abort** command.



### CAUTION

All defined zone objects in the defined zone configuration are deleted. If you try to commit the empty defined zone configuration while a zone configuration is enabled, you are warned to first disable the enabled zone configuration or to provide a valid configuration with the same name.

## Examples

To clear the defined zoning database:

```
device(config)# zoning enabled-configuration cfg-action cfg-clear
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

zoning enabled-configuration cfg-action cfg-save, zoning enabled-configuration cfg-namezoning enabled-configuration cfg-name, zoning enabled-configuration cfg-action cfg-save

# zoning enabled-configuration cfg-action cfg-save

Saves the defined configuration to persist across reboots.

## Syntax

**zoning enabled-configuration cfg-action cfg-save**

## Modes

Global configuration mode

## Usage Guidelines

This command writes the defined configuration and the name of the defined zone configuration to nonvolatile memory in all switches in the VCS Fabric.

The saved configuration is automatically reloaded at power on. If a configuration was in effect at the time it was saved, the same configuration is reinstalled with an automatic **zoning enabled-configuration cfg-name** command.

The **zoning enabled-configuration cfg-action cfg-save** command validates the effective configuration by performing the same tests as the **zoning enabled-configuration cfg-name** command on enabling the configuration. If the tests fail, an error message is displayed and the configuration is not saved.

This command commits the current transaction. Pending transactions are pushed to nonvolatile memory. Any empty zones or empty configurations are deleted.

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric. Edits can be performed only from the principal R Bridge.

## Examples

The following example saves the current zone configuration:

```
device(config)# zoning enabled-configuration cfg-action cfg-save
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[zoning enabled-configuration cfg-name](#), [zoning enabled-configuration cfg-action cfg-transaction-abort](#)

# zoning enabled-configuration cfg-action cfg-transaction-abort

Aborts a current fabric-wide transaction without committing the transaction.

All changes made since the transaction was started are removed and the zone configuration database is restored to the state before the transaction was started.

## Syntax

```
zoning enabled-configuration cfg-action cfg-transaction-abort
```

## Modes

Global configuration mode

## Usage Guidelines

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric. Edits can be performed only from the principal R Bridge.

## Examples

To abort the current zone transaction:

```
device(config)# zoning enabled-configuration cfg-action cfg-transaction-abort
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[zoning enabled-configuration cfg-action cfg-save](#), [zoning enabled-configuration cfg-namezoning enabled-configuration cfg-action cfg-save](#), [zoning enabled-configuration cfg-name](#)

# zoning enabled-configuration cfg-name

Enables a zone configuration.

This command commits the current defined zone configuration to both volatile and nonvolatile memory.

## Syntax

**zoning enabled-configuration cfg-name** *cfgName*

**no zoning enabled-configuration cfg-name**

## Command Default

Zoning is not implemented and default zoning applies.

## Parameters

*cfgName*

Specifies the configuration to be enabled.

## Modes

Global configuration mode

## Usage Guidelines

Only one configuration can be enabled at a time.

Zoning configuration data are automatically distributed among all RBridges in the VCS Fabric.

This command can be entered on any RBridge in an Extreme VCS Fabric. Edits can be performed only from the principal RBridge.

If the operation fails, the previous state is preserved; that is, zoning remains disabled, or the previous effective configuration remains in effect. If the operation succeeds, the new configuration replaces the previous effective configuration.

Enter **no zoning enabled-configuration cfg-name** to disable the currently enabled configuration. The VCS Fabric returns to default zoning mode. In this mode, either all devices can access one another, or no device can access any other device, depending on if the default zoning mode is ALLACCESS or NOACCESS.

### NOTE

If more than 300 devices are connected when the enabled-configuration is disabled, the **no zoning enabled-configuration cfg-name** command is not allowed if the defzone mode is AllAccess. In this case, change the defzone mode to No Access and then disable the enabled-configuration.

## Examples

To enable a zone configuration:

```
device(config)# zoning enabled-configuration cfg-name myconfig
```

To disable the currently enabled configuration:

```
device(config)# no zoning enabled-configuration cfg-name
```

## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[show running-config zoning defined-configuration](#), [show running-config zoning enabled-configuration](#), [zoning enabled-configuration cfg-action](#) [cfg-transaction-abort](#)

# zoning enabled-configuration default-zone-access

Sets the default zone access mode.

## Syntax

```
zoning enabled-configuration default-zone-access { allaccess | noaccess }
```

## Command Default

Zone access mode is "All Access".

## Parameters

### allaccess

Sets the default zone access mode to "All Access". Each device can access all other devices attached to the VCS Fabric.

### noaccess

Sets the default zone access mode to "No Access". No device can access any other device in the VCS Fabric.

## Modes

Global configuration mode

## Usage Guidelines

Zoning configuration data are automatically distributed among all R Bridges in the VCS Fabric.

This command can be entered on any R Bridge in an Extreme VCS Fabric. Edits can be performed only from the principal R Bridge.

### ATTENTION

Setting the default zone mode initializes a zoning transaction (if one is not already in progress), and creates reserved zoning objects. For the change to become effective, you must commit the transaction with either the **zoning enabled-configuration cfg-action cfg-save** command or the **zoning enabled-configuration cfg-name** command.

### NOTE

A default zone controls the device access that is in effect when zoning is not enabled. When a user-specified zoning configuration is not enabled, the default zone is in effect, allowing access to all devices or no devices. When a user-specified zone configuration is enabled, it overrides the default zone access mode.

## Examples

To set the default zone mode to All Access:

```
device(config)# zoning enabled-configuration default-zone-access allaccess
```



## History

Release version	Command history
7.1.0	This command was modified to remove references to fabric cluster mode.

## Related Commands

[zoning enabled-configuration cfg-name](#), [zoning enabled-configuration cfg-action cfg-save](#)