53-1003831-01
18 August 2015

Brocade NetIron QoS and Traffic Management

Configuration Guide

Supporting Multi-Service IronWare R05.9.00

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

The authors and Brocade Communications Systems, Inc. assume no liability or responsibility to any person or entity with respect to the accuracy of this document or any loss, cost, liability, or damages arising from the information contained herein or the computer programs that accompany it.

The product described by this document may contain open source software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit http://www.brocade.com/support/oscd.

Contents

Preface

Document conventions

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

Text formatting conventions

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

Format	Description
bold text	Identifies command names Identifies keywords and operands Identifies the names of user-manipulated GUI elements Identifies text to enter at the GUI
italic text	Identifies emphasis Identifies variables and modifiers Identifies paths and Internet addresses Identifies document titles
`Courier font`	Identifies CLI output 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
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.

Convention	Description
value	In Fibre Channel products, a fixed value provided as input to a command option is printed in plain text, for example, **--show** WWN.
[ ]	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. In Fibre Channel products, square brackets may be used instead for this purpose.
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, *member*[*member*...].
\	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.

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.

Brocade resources

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

You can download additional publications supporting your product at www.brocade.com. Select the Brocade Products tab to locate your product, then click the Brocade product name or image to open the individual product page. The user manuals are available in the resources module at the bottom of the page under the Documentation category.

To get up-to-the-minute information on Brocade products and resources, go to MyBrocade. You can register at no cost to obtain a user ID and password.

Release notes are available on MyBrocade under Product Downloads.

White papers, online demonstrations, and data sheets are available through the Brocade website.

Contacting Brocade Technical Support

As a Brocade customer, you can contact Brocade Technical Support 24x7 online, by telephone, or by e-mail. Brocade OEM customers contact their OEM/Solutions provider.

Brocade customers

For product support information and the latest information on contacting the Technical Assistance Center, go to http://www.brocade.com/services-support/index.html.

If you have purchased Brocade product support directly from Brocade, use one of the following methods to contact the Brocade Technical Assistance Center 24x7.

Online	Telephone	E-mail
Preferred method of contact for non-urgent issues: • My Cases through MyBrocade • Software downloads and licensing tools • Knowledge Base	Required for Sev 1-Critical and Sev 2-High issues: • Continental US: 1-800-752-8061 • Europe, Middle East, Africa, and Asia Pacific: +800-AT FIBREE (+800 28 34 27 33) • For areas unable to access toll free number: +1-408-333-6061 • Toll-free numbers are available in many countries.	support@brocade.com Please include: • Problem summary • Serial number • Installation details • Environment description

Brocade OEM customers

If you have purchased Brocade product support from a Brocade OEM/Solution Provider, contact your OEM/Solution Provider for all of your product support needs.

- OEM/Solution Providers are trained and certified by Brocade to support Brocade® products.
- Brocade provides backline support for issues that cannot be resolved by the OEM/Solution Provider.

- Brocade Supplemental Support augments your existing OEM support contract, providing direct access to Brocade expertise. For more information, contact Brocade or your OEM.
- For questions regarding service levels and response times, contact your OEM/Solution Provider.

Document feedback

To send feedback and report errors in the documentation you can use the feedback form posted with the document or you can e-mail the documentation team.

Quality is our first concern at Brocade 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:

- Through the online feedback form in the HTML documents posted on www.brocade.com.
- By sending your feedback to documentation@brocade.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.

About This Document

Supported Hardware and Software

Supported hardware

The following hardware platforms are supported by this release of this guide:

TABLE 1 Supported devices

Brocade NetIron XMR Series	Brocade MLX Series	NetIron CES 2000 and NetIron CER 2000 Series
Brocade NetIron XMR 4000	Brocade MLX-4	Brocade NetIron CES 2024C
Brocade NetIron XMR 8000	Brocade MLX-8	Brocade NetIron CES 2024F
Brocade NetIron XMR 16000	Brocade MLX-16	Brocade NetIron CES 2048C
Brocade NetIron XMR 32000	Brocade MLX-32	Brocade NetIron CES 2048CX
	Brocade MLXe-4	Brocade NetIron CES 2048F
	Brocade MLXe-8	Brocade NetIron CES 2048FX
	Brocade MLXe-16	Brocade NetIron CER 2024C
	Brocade MLXe-32	Brocade NetIron CER-RT 2024C
		Brocade NetIron CER 2024F
		Brocade NetIron CER-RT 2024F
		Brocade NetIron CER 2048C
		Brocade NetIron CER-RT 2048C
		Brocade NetIron CER 2048CX
		Brocade NetIron CER-RT 2048CX
		Brocade NetIron CER 2048F
		Brocade NetIron CER-RT 2048F
		Brocade NetIron CER 2048FX
		Brocade NetIron CER-RT 2048FX

Supported software

For the complete list of supported features and the summary of enhancements and configuration notes for this release, refer to the *Multi-Service IronWare R05.7.00 Release Notes*.

How command information is presented in this guide

For all new content supported in NetIron Release 05.6.00 and later, command information is documented in a standalone command reference guide.

In an effort to provide consistent command line interface (CLI) documentation for all products, Brocade is in the process of completing a standalone command reference for the NetIron platforms. This process involves separating command syntax and parameter descriptions from configuration tasks. Until this process is completed, command information is presented in two ways:

- For all new content supported in NetIron Release 05.6.00 and later, the CLI is documented in separate command pages included in the *NetIron Command Reference*. Command pages are compiled in alphabetical order and follow a standard format to present syntax, parameters, usage guidelines, examples, and command history.

  NOTE
  Many commands from previous NetIron releases are also included in the command reference.

- Legacy content in configuration guides continues to include command syntax and parameter descriptions in the chapters where the features are documented.

If you do not find command syntax information embedded in a configuration task, refer to the *NetIron Command Reference*.

Configuring Traffic Policing for the Brocade NetIron XMR and Brocade NetIron MLX Series

Layer 2 Service Defense

Layer 2 Ethernet networks are commonly found in today Internet Exchange Point and Data Center network. In case of Internet exchange points(IXPs), the flat layer 2 topology is deployed to provide any-to-any connectivity among BGP routers from different ISPs, CSPs and Enterprises, connecting to it. On the other hand, in order to support VM Mobility within and across data centers, layer 2 network in form of VLAN or VPLS is deployed in Data Centers to extend the layer 2 network from one server to another server within the Data Center or from the Data Center LAN network over the WAN network to the other Data Center LAN network .

As the scale of these flat layer 2 networks grow, without counter-measure to protect the network against abusive or malfunction hosts connecting to it, the security and stability of the network can be easily compromised and jeopardized. The layer 2 defense feature set is to plug the security or operational loophole commonly found in the large scale flat layer 2 network. The feature consists of ARP Guard and BUM rate-limit to address two major challenges in deploying such networks.

ARP Guard

Internet exchange points (IXPs) are designed based on the flat layer 2 topology to provide any-to-any connectivity among BGP routers from different ISPs, CSPs and Enterprises, connecting to it.

As an IP host, each BGP peering router makes use of ARP protocol to determine the MAC address of its BGP peers. Since ARP is not a secure protocol, any BGP router can reply to the ARP request for any IP address and any BGP router can generate gratuitous ARP to claim itself to be the owner of any IP address in the router.

When the network administrator of a BGP border router connecting to the IXP wrongly configures the router IP address or unknowingly turns on the proxy-ARP feature on the interface facing the IXP, this may cause valid traffic to be destined to wrong destination on the wrongly configured BGP border router until ARP cache expiry on the other routers.

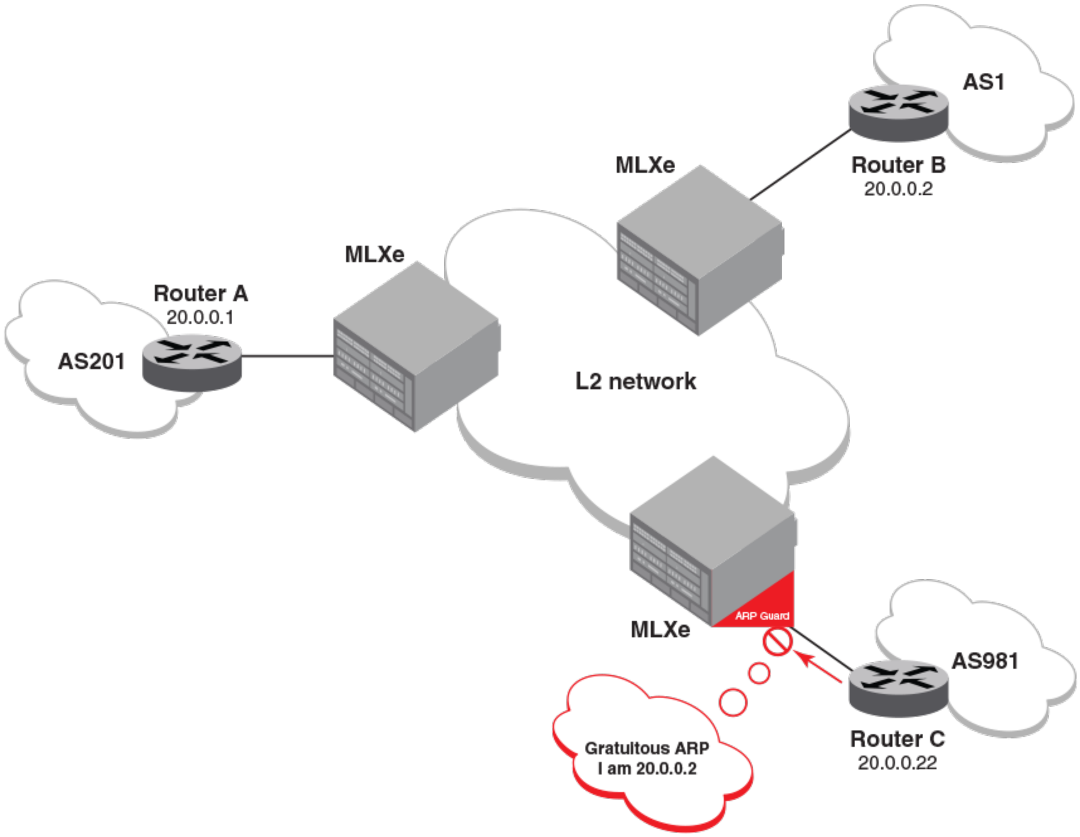
The ARP guard feature uses an ACL-like, CLI parameters (which include VLAN ID, source MAC address and source IP address) to build a table of allowed IP addresses on the link on which this feature is enabled. So, when an ARP reply (either due to gratuitous ARP or normal ARP reply, when proxy-ARP is enabled) arrives at our box on a port facing the BGP router, the ARP packets will be inspected based on the IP address parameter configured using **permit** command. Those ARP packets that do not match the entries in the ACL will be dropped and those which match will be forwarded based on normal forwarding routines.

ARP guard use case scenarios

ARP Hijacking due to wrong configuration of IP address

The network diagram explains how ARP guard functions when IP is wrongly configured on the network.

Layer 2 network with ARP guard:



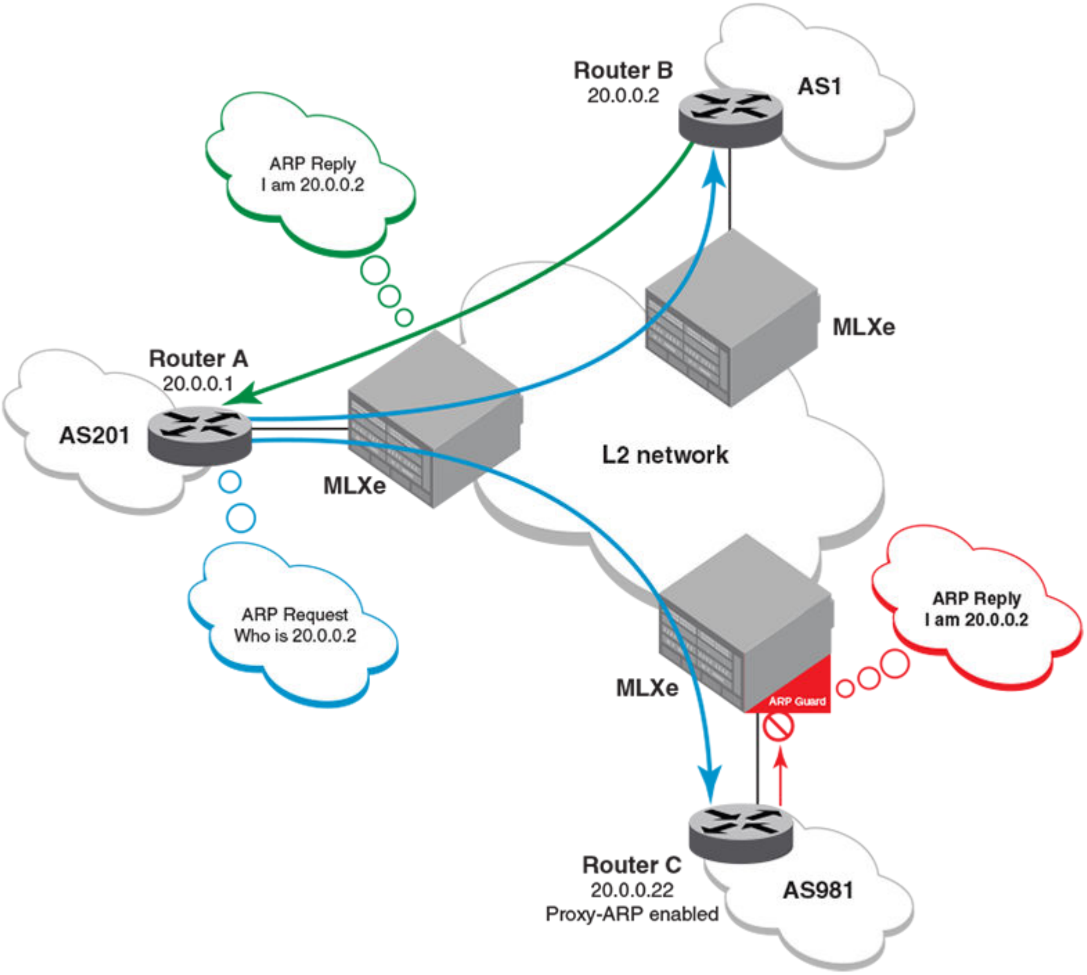
In the above diagram, assume that the correct IP address of router C is 20.0.0.22 and the network administrator of router C mis-configured the router IP address as 20.0.0.2 (which happens to be th IP address of router B from AS 1).

After entering the wrong IP address or after the link from router C to Brocade MLXe Series comes up, router C sends out gratuitous ARP to claim itself to be 20.0.0.2. Without the ARP guard feature, router A may update its ARP entry for 20.0.0.2 with the MAC address of router C causing traffic originally destined to router B to be black-holed on router C. With the ARP guard feature on Brocade MLXe Series, Brocade MLXe Series is configured to just allow gratuitous ARPs for 20.0.0.22 from the link connecting to router C to enter the VLAN or VPLS L2 Network. Upon receiving gratuitous ARP for 20.0.0.2 from router C, it will drop the ARP packet and/or log a message.

ARP Hijacking due to enabling proxy-ARP

The network diagram explains how ARP guard functions when proxy-ARP is enabled on the network.

Layer 2 network with ARP guard:



In the diagram above, assume that the network administrator of router C does configure its IP address correctly to 20.0.0.22 but unknowingly turns on the proxy-ARP. When router A tries to resolve the MAC address for the IP address 20.0.0.2 on router B through ARP, router C will send the ARP reply claiming itself to be the owner of the IP address 20.0.0.2.

Without the ARP guard feature, router A may mistake the MAC address of router C as the MAC address for the IP address 20.0.0.2 of router B, causing traffic originally destined to router B to be black-holed on router C. With the ARP guard feature on Brocade MLXe Series, the Brocade MLXe Series is configured to just allow ARP replies for 20.0.0.22 from the link connecting to router C to enter the VLAN or VPLS L2 network. Upon receiving ARP reply for 20.0.0.2 from router C, it will drop the ARP packet and/or log a message.

Configuration considerations and limitations of ARP guard

Configuration considerations and limitations of ARP guard:

- The configuration options for ARP guard rules would only be based on "permit" options with the following parameters:
  - VLAN ID
  - Source MAC address
  - Source IP address
- If a MAC access group ACL is used to specify the "arp-guard" keyword to punt the ARP packets to CPU, then the incoming traffic cannot be rate-limited due to existing limitation on co-existence of multiple L2 ACLs on the same interface; hence high CPU conditions may arise if the incoming rate of ARP traffic is more.
- Recommend usage of Rate Limit L2 ACL if rate limiting is to be done on the associated interface where ARP Guard is to be turned ON.
- ARP guard is supported only on physical interfaces.
- Cannot configure ARP guard if the physical interface has IPv4 related configurations and vice versa.
- Cannot configure ARP guard on a route-only interface and vice versa.
- ARP guard statistics are not retained after a switchover.
- On a LAG that has ARP guard enabled, if a member port is removed from the LAG, then the ARP guard properties will still be retained on the removed port of that LAG since it becomes an individual physical port.
- The **show-run** command will not display the ARP guard configurations with default conditions.
- Cannot configure global route-only option if ARP guard is enabled on any of the interfaces.
- Support for **show** command for ARP guard exists only in MP.

Configuring ARP guard

Following commands configure ARP guard on a Brocade device.

Enabling filtering of incoming ARP packets to LP-CPU:

The following command enables the required L2 ACL rules to filter the incoming ARP packets to the CPU. In the following option, an additional key word "arp-guard" is supported in the existing L2 access-list command syntax. The user should specify this key word when creating the L2 ACL rules for filtering the ARP packets to CPU.

```
Brocade(config)# access-list 400 permit any any any etype arp arp-guard
```

Syntax: [no] access-list*num* **permit** *src-mac / mask* | **any** *dest-mac*|*mask* | **any** *vlan-id* | **any etype arp arp-guard**

This configuration creates a standard Layer-2 ACL with an ID of 400.

etype arp : L2 ACL applied only for the ARP packets.

arp-guard : ARP-Guard will filter all the ARP packets to the LP-CPU.

Bind Layer 2 ACL to an interface:

Layer 2 ACL needs to be bound to an interface where ARP guard would be required. The below configuration will punt all incoming ARP packets to LP-CPU based on the L2-ACL rules provided.

```
Brocade(config-if-e10000-1/1)# mac access-group 400 in
```

NOTE

The keyword "arp-guard" is mandatory for Brocade NetIron MLX Series and Brocade NetIron XMR Series series of device to handle the programming of the ARP guard ACLs in hardware in order to punt the received ARP packets to LP CPU for processing.

For Brocade NetIron CER Series and Brocade NetIron CES Series platforms, by default all ARP packets are trapped to CPU. Hence, the keyword "arp-guard" is not required to handle the programming of ARP guard ACLs in hardware. Also the binding of ACLs to the associated interface is not required in Brocade NetIron CER Series and Brocade NetIron CES Series devices.

Creating ARP guard access-list table:

The rules for filtering of ARP packets are done through ACL like commands in the global configuration mode, which are configured through the following commands. The **no** form of the command would disable that particular rule.

```
Brocade(config)# arp-guard-access-list AS201
Brocade(config-arp-guard-access-list-AS201)# permit 1.1.1.1
Brocade(config-arp-guard-access-list-AS201)# permit 1.1.1.2 1111.1111.1111
Brocade(config-arp-guard-access-list-AS201)# permit 1.1.1.3 any
Brocade(config-arp-guard-access-list-AS201)# permit 10 1.1.1.4
Brocade(config-arp-guard-access-list-AS201)# permit 10 1.1.1.4 1111.2222.3333
```

Syntax: [no] arp-guard-access-list *arp-guard-access-list*

Syntax: [no] permit *[vlan-id][src-ip-addr] [src-mac-addr]*|**[any]**

Parameters

arp-guard-access-list specifies the name of the ARP guard access-list.

permit specifies the required set of rules for the associated ARP guard group.

Binding of ARP guard to an interface:

The following commands are used to enable ARP guard under the interface configuration mode. Using "log" option, would capture the log information of the dropped ARP packet such as "the name of the port", "vlan-id"(if any), "name of the ACL" which detected the violation, "MAC-address", and "IP address". The **no** form of the command would disable ARP guard.

```
Brocade(config-if-e10000-1/1)# arp-guard AS201
Brocade(config-if-e10000-1/1)# arp-guard AS201 log
Brocade(config-if-e10000-1/1)# arp-guard AS201 log 20
```

Syntax: [no] arp-guard *arp-guard-access-list* **[log]***number of violations to cache*

Parameters

arp-guard enables ARP guard in the interface configuration mode.

arp-guard-access-list specifies the name of the ARP guard access-list which contains the list of rules.

log option is used to log the information about the dropped packets.

number of violations to cache specifies the number of dropped packets to cache. Range is 5 to 32.

NOTE

If user does not specify number of violations, then by default; last 5 violated dropped packets will be printed on the active console at every default interval or configured interval.

Modifying ARP guard rules:

If user modifies an existing bound ARP-Guard access-list, then **apply-arp-guard** command should be used to apply the changed rules on the associated interfaces.

```
Brocade(config-arp-guard-access-list-AS201)#apply-arp-guard
```

Syntax: apply-arp-guard

apply-arp-guard will program all the newly updated rules(if any) for that session(console/telnet/SSH) to all the associated ports.

NOTE
Invalid rules will be discarded in the following scenarios.

- When user jumps into different prompt from the ARP-Guard prompt.
- When user triggers “end”/”exit” from the ARP-Guard prompt.
- When the current session is closed (telnet/SSH).
- When the active MP switches over to the standby MP.

Steps to configure ARP guard:

1. Configure L2-ACL rules for ARP packet filtering to LP-CPU. The example below uses MAC ACLs, even RL ACLs can be used for the same.

```
Brocade(config)#access-list 400 permit any any any etype arp arp-guard
Brocade(config-if-e1000-1/24)#mac access-group 400 in
```

2. Configure arp-guard-access-list to specify the set of rules/filters for this ARP ACL.

```
Brocade(config)#arp-guard-access-list AS201
Brocade(config-arp-guard-access-list)#permit 20.0.0.2 0001.0002.0003
Brocade(config-arp-guard-access-list)#exit
```

3. Apply the arp-guard-access-list on the interface using the arp-guard command as shown below.

```
Brocade(config)#interface ethe 1/1
Brocade(config-if)#arp-guard AS201 log
```

Syslog Information

If **log** option is specified in the **arp-guard** command, then a syslog message is generated to log the dropped ARP packet. The **arp-guard-syslog-timer** command can be used to modify the interval at which the syslogs need to be generated

Syslog message contains the following:

- Port ID
- arp-guard-group name
- VLAN-id (if any)
- MAC address and the IP address

All violations are noted down in Software and at the configured syslog interval for the ARP Guard entry the violations are logged.

Following are the Syslog message output display:

```
SYSLOG: <14>Mar 14 1905 22:37:21 MLX-Dist1 ARP_GUARD DROP LOG:10 Violations occurred
on port=4/1 having Access_Grp= AS201 Most recent 5 violations are:

SYSLOG: <14>Mar 14 1905 22:37:21 MLX-Dist1 ARP_GUARD DROP LOG:Violation occured at
time Mar 14 22:37:20: on Trunk port=4/1 having Access_Grp=AS201, for the incoming
packet with MAC_ADDR=0000.5822.bf78 IP_ADDR=1.1.1.2 VLAN: 1

SYSLOG: <14>Mar 14 1905 22:37:21 MLX-Dist1 ARP_GUARD DROP LOG:Violation occured at
time Mar 14 22:37:20: on Trunk port=4/1 having Access_Grp= AS201, for the incoming
```

```
packet with MAC_ADDR=0000.5823.0a9b IP_ADDR=2.1.1.2 VLAN: 1

SYSLOG: <14>Mar 14 1905 22:37:21 MLX-Dist1 ARP_GUARD DROP LOG:Violation occured at
time Mar 14 22:37:20: on Trunk port=4/1 having Access_Grp= AS201, for the incoming
packet with MAC_ADDR=0000.5822.bf78 IP_ADDR=1.1.1.2 VLAN: 1

SYSLOG: <14>Mar 14 1905 22:37:21 MLX-Dist1 ARP_GUARD DROP LOG:Violation occured at
time Mar 14 22:37:20: on Trunk port=4/1 having Access_Grp= AS201, for the incoming
packet with MAC_ADDR=0000.5823.0a9b IP_ADDR=2.1.1.2 VLAN: 1
```

Rate-limiting BUM packets

To prevent the flat layer 2 network such as VPLS and VLAN from being flooded with excessive amount of broadcast, unknown-unicast, and multicast (a.k.a BUM) traffic, the BUM traffic received on a port can be restricted using BUM rate-limiter. With BUM rate-limit configured on the port, when a high rate of BUM traffic is received on the port, BUM traffic will be throttled to the configured data rate associated with the BUM rate-limit before being replicated across the Layer 2 network.

When the received BUM traffic exceeds the pre-defined rate limit, the physical port can be configured to be automatically shutdown or disable using the shutdown option of the **rate-limit** command. The port shut down occurs within 2.5 seconds after the BUM traffic exceeds the defined limit. The port can be enabled again through the use of the **clear** command.

Limitations of BUM rate-limiting

BUM rate-limiting has the following limitations:

- The BUM rate-limit does not coexist with the VPLS-CPU-protection feature.
- VLAN based rate-limiting and port based rate-limiting can co-exist. However, when a VLAN is shutdown, only the ingress BUM traffic will be blocked. For Port based rate-limiting, Shutdown disables the port.
- ARP traffic is sent to the CPU without rate-limiting unless a specific ARP rate-limiting command is used to configure ARP rate-limiting.
- Control packets are not rate limited.
- It cannot coexist with any of the flooding features like transparent VLAN flooding/unknown unicast flooding.
- If ACL is applied, it has the highest priority to block the traffic than BUM rate-limiting.
- When BUM rate-limit is configured with multicast and/or alert option, and if there is L2/L3 traffic present then alert and shutdown will occur only when the traffic is being sent to CPU. Once the hardware entries are created (Mcache or Snooping) traffic will be rate-limited but alert and shutdown will not occur.
- Gen 1.1 and 24x10GbE–DM card do not support **alert** and **shutdown** option due to hardware limitation.
- Since BUM rate-limit is a one-time application command, if the **include-control** option is not enabled in that command, another similar BUM rate-limit cannot be configured.
- Gen 1 and Gen 1.1 cards support the **include-control** option; however, ARP packets will not be rate-limited.
- Only those control packets that match the ACL filter or the DA match for BUM rate-limit will be rate-limited.
- Only Gen 2 cards that have FPGA support will support ARP packets for rate-limit.
- When any protocol is enabled and a corresponding CAM entry is programmed, those control packets will not be rate-limited. This is due to its presence above the flooding partition.
- Protocols such as LACP, STP, MRP, and VRRP, if enabled, will not be rate-limited through BUM rate-limit. However, they can be rate-limited through the ACL based rate-limiting.

Configuring per-port rate limiting for BUM traffic

To configure per-port rate limiting for BUM traffic on Brocade NetIron XMR Series and Brocade NetIron MLX Series device:

1. Run the **interface ethernet** *slot/port number* command to enter the interface mode.
2. Run the **rate-limit input {broadcast | unknown-unicast | multicast}** command to configure the average-rate, maximum burst size, shutdown timeout, and alert functions for BUM traffic rate limiting on a specific port.

Syntax: rate-limit input {broadcast | unknown-unicast | multicast} *[average-rate* | *max burst size]* **include-control | shutdown [timeout] alert** *[high-water-mark* | *low-water-mark]*

The **input** parameter applies the rate limiting policy to traffic on inbound ports.

The **broadcast, unknown-unicast, and multicast** parameters define a rate limit for ingress broadcast, unknown-unicast, and multicast packets on the port. Any combination of these parameters can be used to define the rate limit.

The *average-rate* variable specifies the maximum number of bits a port is allowed to receive during a one-second interval and is the aggregate sum of the broadcast, unknown-unicast, and multicast packets rate limit, if the rate limit is configured for all three packets. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps).

The *max burst size* variable specifies the value of the maximum burst of traffic allowed by the specific port.

The **shutdown** option specifies that the port is to be shut down if the amount of BUM traffic exceeds the pre-defined limit. Time out value is between 0 to1440 minutes.

NOTE
Setting the timeout value as **0 (zero)** disables the port until it is manually enabled.

The **include-control** option extends the Service Defense functionality to include ARP and other control packets under existing ACL-based and BUM rate -limit policies.

The **include-contro**l option enables the Quality of Service (QoS) Override Flag on the PPCR. Therefore, the specific stream that gets filtered due to ACL/BUM DA match alone gets rate-limited.

For example: if the ACL used is a Generic L2 ACL with Source MAC : aabb:ccdd:eeff, then any control packet received from this MAC address will be included in the rate limit bucket.

The **alert** option alerts if the traffic exceeds or fall short of the defined limit on the specific port. The alert option can be defined with high watermark value or with low watermark value in bps.

The following example shows how to configure the rate-limit on a specific port with average-rate, max burst size, shutdown, and alert function defined.

```
Brocade (config-if-e1000-1/1)#rate-limit input broadcast unknown-unicast
multicast 100000 15000 include-control shutdown 1 alert 80000 10000
```

Syslog

Upon receipt of excessive BUM traffic, BUM rate-limit can be configured to bring down the port. When this happens, a log is generated. Another log message will be generated on the expiry of the auto-shutdown timer.

NOTE
The Syslog messages are printed continuously for smallest rate-limit values with (average rate-limit value 8144 and burst rate 10000), and very small values of high and low watermark for alert option with (high watermark value 1000 and low watermark value 23).

Following is the Syslog message output:

```
Brocade#show log
Ethernet 1/1 VLAN 100 is blocked due to  broadcast rate-limit threshold auto-
shutdown: action = disabled, state = down
Ethernet 1/1 is unblocked  due to broadcast rate-limit threshold auto-shutdown timer
expiry: action = enabled, state = up
```

Configuring port per VLAN rate limiting for BUM traffic

To configure port per VLAN rate limiting for BUM traffic on Brocade NetIron XMR Series and Brocade NetIron MLX Series device:

1. Run the **interface ethernet***slot/port number* command to enter interface mode.
2. Run the **rate-limit input {broadcast | unknown-unicast | multicast}** command to configure the average-rate, maximum burst size, shutdown timeout, and alert functions for BUM traffic rate limiting on a VLAN.

Syntax: rate-limit input {broadcast | unknown-unicast | multicast} vlan-id *[vlan-id][average-rate | max burst size]* **shutdown [timeout] alert** *[high-water-mark | low-water-mark]*

The **input** parameter applies the rate limiting policy to traffic on inbound ports.

The **broadcast, unknown-unicast, and multicast** parameters define a rate limit for ingress broadcast, unknown-unicast, and multicast packets on the port. Any combination of these parameters can be used to define the rate limit.

The **vlan-id** specifies the VLAN id of the specific port on which the rate-limiting of BUM traffic is accounted.

The *average-rate* variable specifies the maximum number of bits a port is allowed to receive during a one-second interval and is the aggregate sum of the broadcast, unknown-unicast, and multicast packets rate limit, if the rate limit is configured for all three packets. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second bps.

The *maximum burst size* variable specifies the value of the maximum burst of traffic allowed by the specific port.

The **shutdown** option specifies that the port is to be shut down if the amount of BUM traffic exceeds the pre-defined limit. Time out value is between 0 to1440 minutes.

NOTE
Setting the timeout value as **0 (zero)** disables the port until it is manually enabled.

The **alert** option alerts if the traffic exceeds or fall short of the defined limit on the specific port. The alert option can be defined with high watermark value or with low watermark value in bps.

The following example shows how to configure the rate-limit on a VLAN with average-rate, max burst size, shutdown, and alert function defined.

```
Brocade (config-if-e1000-1/1)#rate-limit input broadcast unknown-unicast
multicast vlan-id 200 100000 15000 shutdown 1 alert 80000 10000
```

Displaying per-port / per VLAN rate limiting information for BUM traffic

Run the **show rate-limit counters bum-drop[all] [port-id]***[slot/port]***[vlan-id]** *vlan* command to display the per-port / per VLAN rate limiting information for BUM traffic.

Syntax

show rate-limit counters bum-drop[all] [port-id]*[slot/port]***[vlan-id]** *vlan*

Description

port-id specifies the port number for which the rate-limiting is configured.

vlan-id specifies the VLAN for which the rate-limiting is configured.

Command Output

The **show rate-limit counters bum-drop** *[port-id] [vlan-id]* command displays the following information:

Output field	Description
interface	Displays information about the interface information for which the rate-limiting accounting is configured.
port: Drop:	Displays information about the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy for the specific port defined.
rate-limit input broadcast	Displays information about the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy.
vlan-id: 100 Drop	Displays information about the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy for the specific VLAN id defined.

The following example for **show rate-limit counters bum-drop** command displays the following information:

```
Brocade (config-if-e10000-5/1)#sh rate-limit counters bum-drop

interface e 5/1
rate-limit input broadcast 993568 10000
port: Drop: 0 bytes
rate-limit input vlan-id 100 broadcast 993568 100000
vlan-id: 100 Drop: 0 bytes

Brocade (config-if-e10000-5/1)#sh rate-limit counters bum-drop port-id 5/1

interface e 5/1
rate-limit input broadcast 993568 10000
port: Drop: 0 bytes

Brocade (config-if-e10000-5/1)#sh rate-limit counters bum-drop port-id 5/1 vlan-id 100

interface e 5/1
rate-limit input vlan-id 100 broadcast 993568 100000
vlan-id: 100 Drop: 0 bytes
```

Clearing the per-port/per VLAN rate limiting information for BUM traffic

Run the following command to clear the accounting information for BUM traffic.

Run the **clear rate-limit counters bum-drop[shutdown] [port-id]** *slot/port***[all] [vlan-id]** *[vlan]*

Syntax

clear rate-limit counters bum-drop[shutdown] [port-id] *slot/port***[all] [vlan-id]** *[vlan]*

Description

port-id specifies the port number for which the rate-limiting is configured.

vlan-id specifies the VLAN for which the rate-limiting is configured.

shutdown option clears the shutdown status of the corresponding port/VLAN for BUM rate-limiting.

Commands

The following commands support the features described in this chapter:

- rate-limit input multicast broadcast unknown-unicast
- show rate-limit counters bum-drop
- clear rate-limit counters bum-drop
- snmp-server enable traps bum-rl-traps

Rate-limit input

Configures the per-port or port per VLAN BUM rate-limiting.

Syntax **rate-limit input** [**vlan** *vlan id*] [**broadcast** | **unknown-unicast** | **multicast**][*average-rate maximum burst size*][**include-control**][**shutdown** *timeout*][**alert** *high-watermark low-watermark*]

Parameters **broadcast | unknown-unicast | multicast**

The option define a rate limit for ingress broadcast, unknown-unicast, and multicast packets on the port. Any combination of these parameters can be used to define the rate limit.

vlan

Specifies the VLAN id of the specific port on which the rate-limiting of BUM traffic is accounted.

average-rate

Specifies the maximum number of bits a port is allowed to receive during a one-second interval and is the aggregate sum of the broadcast, unknown-unicast, and multicast packets rate limit, if the rate limit is configured for all three packets. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps).

maximum burst size

Specifies the value of the maximum burst of traffic allowed by the specific port.

include-control

Extends the existing BUM rate-limit to include rate limit of ARP, other control packets.

shutdown

Specifies that the port is to be shut down if the amount of BUM traffic exceeds the pre-defined limit. Time out value is between 0 to 1440 minutes.

alert

Alert message if the rate crossed over/under limit shutdown. Shut down the port if the rate is over limit.

Modes Interface configuration mode

Usage Guidelines

Examples The following is an example for rate-limit input configuration.

```
device(config)#int eth 1/1
device(config-if-e1000-1/1)#rate-limit input broadcast 100000 10000 include-control
shutdown 1 alert 80000 10000
device(config-if-e1000-1/1)#rate-limit input multicast 100000 10000 include-control
shutdown 1 alert 80000 10000
```

History

Release version	Command history
Release 05.7.00	This command was introduced.
Release 05.9.00	This command was modified to include the **include-control** option.

Rate-limit input access-group

Configures input access group rate-limiting.

Syntax **rate-limit input access-group** [*group-number*|**name** {**ipv4**| **ipv6** | **mac** }*name*][*average-rate maximum burst size*|**policy-map** *map-name*][**include-control**]

Parameters **access-group**

Specifies the group-number.

ipv4 | ipv6 | mac

Specifies the address type.

average-rate

Specifies the maximum number of bits a port is allowed to receive during a one-second interval and is the aggregate sum of the broadcast, unknown-unicast, and multicast packets rate limit, if the rate limit is configured for all three packets. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps).

maximum burst size

Specifies the value of the maximum burst of traffic allowed by the specific port.

policy-map

Specifies that the policy-map name.

include-control

Extends the existing BUM rate-limit to include rate limit of ARP, other control packets.

Modes Interface configuration mode

Examples The following is an example for rate-limit input access-group configuration.

```
device(config)#int eth 1/1
device(config-if-e1000-1/1)#rate-limit input access-group 400 100000 10000 include-
control
device(config-if-e1000-1/1)#rate-limit input access-group 1 100000 10000 include-
control
device(config-if-e1000-1/1)#rate-limit input access-group 100 100000 10000 include-
control
```

History

Release version	Command history
Release 05.7.00	This command was introduced.
Release 05.9.00	This command was modified to include the **include-control** option.

show rate-limit counters bum-drop

Displays the per-port / per-VLAN rate-limiting information for broadcast/unicast/multicast (BUM) traffic.

Syntax

show rate-limit counters bum-drop

show rate-limit counters bum-dropport-id *slot* / *port* [**all** | **vlan** *vlan-id*]

Parameters

port-id *slot* / *port*

Displays the information for a specified port.

all

Displays the information for all BUM counters on the specified port.

vlan *vlan-id*

Displays the information for all BUM counters on the specified VLAN.

Modes

User EXEC mode

Command Output

The **show rate-limit counters bum-drop** command displays the following information:

Output field	Description
interface	Displays the interface information for which the rate-limiting accounting is configured.
port: Drop:	Displays information about the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy for the specific port defined.
rate-limit input broadcast	Displays information about the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy.
vlan-id: 100 Drop	Displays information about the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy for the specific VLAN id defined.

Examples

The following example for **show rate-limit counters bum-drop** command displays the following information.

```
Brocade(config-if-e10000-5/1)#sh rate-limit counters bum-drop

interface e 5/1
rate-limit input broadcast 993568 10000
port: Drop: 0 bytes
rate-limit input vlan-id 100 broadcast 993568 100000
vlan-id: 100 Drop: 0 bytes

Brocade(config-if-e10000-5/1)#sh rate-limit counters bum-drop port-id 5/1

interface e 5/1
rate-limit input broadcast 993568 10000
port: Drop: 0 bytes

Brocade(config-if-e10000-5/1)#sh rate-limit counters bum-drop port-id 5/1 vlan-id 100

interface e 5/1
rate-limit input vlan-id 100 broadcast 993568 100000
vlan-id: 100 Drop: 0 bytes
```

History

Release version	Command history
5.7.00	This command was introduced.

clear rate-limit counters bum-drop

Clears the accounting information for the Broadcast, Unicast, Multicast (BUM) traffic rate limit.

Syntax

clear rate-limit counters bum-drop [*portid*] [*vlanid*]

clear rate-limit counters bum-drop [**shutdown**] [*portid*] *slot*/*port* [**all**] [*vlan-id*] [*vlan*]]

Parameters

portid

Optionally clears the accounting information for BUM rate-limiting for the specified port.

vlanid

Optionally clears the accounting information for BUM rate-limiting for the specified VLAN.

Modes

Privileged EXEC configuration mode

Usage Guidelines

This command is used to clear rate-limiting accounting information for BUM traffic and, optionally, for specified interfaces or VLANs.

Examples

The following example clears the BUM rate-limiting information for VLAN 2.

```
device# clear rate-limit counters bum-drop vlan2
```

History

Release version	Command history
5.7.00	This command was introduced.

snmp-server enable traps bum-rl-traps

Configures the SNMP rate-limiting traps for BUM traffic on SNMP servers.

snmp-server enable traps bum-rl-traps

no snmp-server enable traps bum-rl-traps

Command Default

By default, SNMP rate-limiting traps for BUM traffic on SNMP servers are enabled.

Usage Guidelines

Examples

The following example shows how to disable SNMP rate-limiting traps for BUM traffic.

```
device# configure terminal
device(config)# no snmp-server enable traps bum-rl-traps
```

History

Release version	Command history
5.7.00	This command was introduced.

Traffic policing on the Brocade device

The Brocade device provides line-rate traffic policing in hardware on inbound ports and outbound ports.

You can configure a Brocade device to use one of the following modes of traffic policing policies:

- **Port-based** - Limits the rate on an individual physical port to a specified rate. Only one inbound and one outbound port-based traffic policing policy can be applied to a port. (Refer to Configuring traffic policing on Brocade devices on page 35.) These policies can be applied to inbound and outbound traffic.

NOTE
The MLX series does not support BUM rate limiting on a per port level.

- **Port-and-priority-based** - Limits the rate on an individual hardware forwarding queue on an individual physical port. Only one port-and-priority-based traffic policing policy can be specified per priority queue for a port. (Refer to Configuring traffic policing on Brocade devices on page 35.) These policies can be applied to inbound and outbound traffic.
- **VLAN-based** - Untagged packets as well as tagged packets can be rate-limited. Only one rate can be specified for each VLAN. (Refer to Configuring traffic policing on Brocade devices on page 35.) Up to 990 VLAN-based policies can be configured for a port under normal conditions or 3960 policies if priority-based traffic policing is disabled as described in Configuring for no priority-based traffic policing on page 41. These policies can be applied to inbound and outbound traffic.
- **VLAN group based** - Limits the traffic for a group of VLANs. Members of a VLAN group share the specified bandwidth defined in the traffic policing policy that has been applied to that group. (Refer to Configuring traffic policing on Brocade devices on page 35.) Up to 990 VLAN Group-based policies can be configured for a port under normal conditions or 3960 policies if priority-based traffic policing is disabled as described in Configuring for no priority-based traffic policing on page 41. These policies can only be applied to inbound traffic.

NOTE
If a VLAN based policing is configured on a port for a particular VLAN, the policing will be applicable to all ports on that Network Processor that belong to that VLAN.

- **Port-and-ACL-based** - Limits the rate of IP traffic on an individual physical port that matches the permit conditions in IP Access Control Lists (ACLs). Layer 2 ACL-based traffic policing is supported. You can use standard or extended IP ACLs. Standard IP ACLs match traffic based on source IP address information. Extended ACLs match traffic based on source and destination IP address and IP protocol information. Extended ACLs for TCP and UDP also match on source and destination TCP or UDP addresses. and protocol information. These policies can be applied to inbound and outbound traffic. Up to 990 Port-and-ACL-based policies can be configured for a port under normal conditions or 3960 policies if priority-based traffic policing is disabled as described in Configuring for no priority-based traffic policing on page 41.
- **Rate Limiting for Copied-CPU-bound Traffic** - You can limit the rate of Copied-CPU-bound packets from applications such as sFlow, ACL logging, RPF logging, and source MAC address learning (with known destination address). Copied-CPU-bound packets are handled and queued separately from packets destined to the CPU such as protocol packets and using this feature they can be assigned to one of eight priority queues which has a rate limit assigned to it. The queue and rate are assigned by port and apply to all of the ports that are supported by the same packet processor. Configuring rate limiting for Copied-CPU-bound traffic on page 41 describes the ports that are associated a packet processor.

Multi-Service IronWare supports applying traffic policing parameters directly to a port or creating a policy map to define a set of traffic policing parameters and then applying that policy map to one or more ports. In addition, the traffic policing parameters available from each of these options are different. The parameters used when applying traffic policing parameters directly to a port reflect the Multi-Service IronWarefeatures that were available before this release. These parameters and the information required to use them are described in Applying traffic policing parameters directly to a port on page 32.

The parameters used when applying traffic policing through use of a policy map reflect the traffic policing features that have been added with this release. These parameters and the information required to use them are described in Applying traffic policing parameters using a policy map on page 33.

Applying traffic policing parameters directly to a port

When applying a traffic policing policy directly to a port, there are specific parameters that are applied to implement the policy that are different than those used when using a policy map. The Brocade NetIron XMR supports this mode in addition to policy maps. Using this method, a traffic policing policy specifies two parameters: average rate and maximum burst. These parameters are used to configure credits and credit totals.

Average rate

The average rate is the maximum number of bits a port is allowed to receive during a one-second interval. The rate of the traffic that matches the traffic policing policy will not exceed the average rate.

The average rate represents a percentage of an interface's line rate (bandwidth), expressed in bits per second (bps). It cannot be smaller than 8,144 bits per second (bps) and it cannot be larger than the port’s line rate.

For Brocade NetIron MLX Series and Brocade NetIron XMR devices, the average rate must be entered in multiples of 8,144 bps. If you enter a number that is not a multiple of 8,144, the software adjusts the rate down to the lowest multiple of the number so that the calculation of credits does not result in a remainder of a partial Credit. For example, if you enter 10,000 bps, the value will be adjusted to 8,144 bps. The adjusted rate is sometimes called the adjusted average rate .

For Brocade NetIron CER and Brocade NetIron CES devices, the average rate can be entered in as any value from 0 up to the line rate of the port. Multiples of 8,144 do not need to be used.

Maximum burst

Maximum burst provides a higher than average rate to traffic that meet the rate limiting criteria. Traffic will be allowed to pass through the port for a short period of time. The unused bandwidth can be accumulated up to a maximum of "maximum burst" value expressed in bits.

Maximum burst size is adjusted according the configured average line rate. If the user configured maximum burst size value exceeds the maximum burst size allowed, the maximum burst size will be automatically adjusted to the values indicated in the Maximum Burst Size table below.

TABLE 2 Maximum Burst Size

Average rate (bps)	Maximum burst size (Bits)
1 Mbps	66,535
1 - 10 Mbps	524,280
10 - 100 Mbps	4,194,240
100 Mbps - 1 Gbps	33,553,920
1 Gbps - 10 Gbps	268,431,230

Credits and credit total

Each rate limiting policy is assigned a class. A *class* uses the average rate and maximum burst in the rate limit policy to calculate credits and credit totals.

Credit size is measured in bytes. A credit is a forwarding allowance for a traffic policed port, and is the smallest number of bytes that can be allowed during a rate limiting interval. Minimum credit size can be 1 byte.

During a rate limiting interval, a port can send or receive only as many bytes as the port has Credits for. For example, if an inbound rate limiting policy results in a port receiving two credits per rate limiting interval, the port can send or receive a maximum of 2 bytes of data during that interval.

In each interval, the number of bytes equal to the credit size is added to the running total of the class. The running total of a class represents the number of bytes that can be allowed to pass through without being subject to rate limiting.

The second parameter is the maximum credit total, which is also measured in bytes. The maximum credit total is based on the maximum burst value and is also measured in bytes.

The running total can never exceed the maximum credit total. When packets arrive at the port, a class is assigned to the packet based on the traffic policing policies. If the running total of the class is less than the size of the packet, then the packet is dropped. Otherwise, the size of the packet is subtracted from the running total and the packet is forwarded. If there is no traffic that matches traffic policing criteria, then the running total can grow up to the maximum credit total.

Applying traffic policing parameters using a policy map

When using the traffic policing policies available from previous versions, the policy parameters are provided explicitly for each port during port configuration. In this version, the policies must be defined using a policy map. The policy map configuration ties a policy name to a set of traffic policing policies. The policy name is then applied to the port or ports that you want to rate limit using the defined policy. This allows you to set a policy in a single location the affects multiple ports and to make changes to that policy. Configuration of a policy map is described in Configuring traffic policing on Brocade devices on page 35.

Within the policy map configuration, the parameters used to define traffic policing have been changed. When configuring traffic policing within a policy map, these new parameters apply. With this release, traffic policing policy determines the rate of inbound or outbound traffic (in bits per second or bps) that is allowed per port. This traffic is initially traffic policed by a Committed Information Rate (CIR) bucket. Traffic that is not accommodated in the CIR bucket is then subject to the Excess Information Rate (EIR) bucket.

The CIR bucket

The CIR rate limiting bucket is defined by two separate parameters: the CIR rate, and the Committed Burst Size (CBS) rate. The CIR rate is the maximum number of bits a port is allowed to receive or send during a one-second interval. The rate of the traffic that matches the traffic policing policy can not exceed the CIR rate. The CIR rate represents a portion of an interface's line rate (bandwidth), expressed in bits per second (bps) and it cannot be larger than the port’s line rate. CIR-defined traffic that does not use the CIR rate available to it accumulates credits that it can use later in circumstances where it temporarily exceeds the CIR rate.

When traffic exceeds the bandwidth that has been reserved for it by the CIR rate defined in its policy, it becomes subject to the CBS rate. The CBS rate provides a rate higher than the CIR rate to traffic that exceeded its CIR rate. The bandwidth in the CBS rate, as expressed in bytes, is accumulated during periods of time when traffic that has been defined by a policy does not use the full CIR rate available to it. Traffic is allowed to pass through the port for a short period of time at the CBS rate.

When inbound or outbound traffic exceeds the bandwidth available for the defined CIR and CBS rates, it is either dropped, or made subject to the conditions set in it EIR bucket.

The EIR bucket

The EIR bucket provides an option for traffic that has exceeded the conditions set by policy for the CIR bucket. In the EIR bucket, there are two parameters that define the traffic that is available: the Excess Information Rate (EIR) and the Excess Burst Size (EBS) rate. The EIR and EBS operate exactly like the CIR and CBS except that they only act upon traffic that has been passed to the EIR bucket because it could not be accommodated by the CIR bucket. Like the CIR, the EIR provides an initial bandwidth allocation to accommodate inbound and outbound traffic. If the bandwidth provided by the EIR is insufficient to accommodate the excess traffic, the defined EBS rate provides for burst traffic. Like the CBS, the bandwidth available for burst traffic from the EBS is subject to the amount of bandwidth that is accumulated during periods of time when traffic that has been allocated by the EIR policy is not used.

In addition, to providing additional bandwidth for traffic that exceeds that available for the CIR bucket, traffic rate limited by the EIR bucket can have its excess priority and excess dscp values changed. Using this option, priority parameters are set following the EBS value that change the priority of traffic that is being rate limited using the EIR bucket.

Configuration considerations

- Only one type of traffic policing policy can be applied on a physical port. For example, you cannot apply port-and-ACL-based and port-based traffic policing policies on the same port.
- When a VLAN-based traffic policing policy is applied to a port, all the ports controlled by the same packet processor are rate limited for that VLAN. You cannot apply a VLAN-based traffic policing policy on another port of the same packet processor for the same VLAN ID.
- The Multi-Service IronWare software supports VLAN-based traffic policing that can limit tagged and untagged packets that match the VLAN ID specified in the policy. Untagged packets are not subject to traffic policing.
- The maximum burst in a traffic policing policy cannot be less than the average rate and cannot be more than the port’s line rate.
- Control packets are not subject to traffic policing.
- Source MAC address with Virtual Leased Line (VLL) endpoints are not subject to traffic policing.

Configuring traffic policing on Brocade devices

The following sections show examples of how to configure each traffic policing policy type.

Configuring a policy map

To configure a policy map, enter a command such as the following.

```
device(config)#policy-map map5
device(config-policymap map5)#cir 1000000 cbs 2000000 eir 1000000
ebs 2000000 excess-dp 2 excess-dscp 37
```

The command configures the traffic policing policy map map1 to limit CIR rate to 1000000 bits per second (bps) the CBS rate to 2000000 bits per second (bps), the EIR rate to 1000000 bits per second (bps) and the EBS to 2000000 bits per second (bps). In addition, traffic that exceeds the bandwidth available in the CIR bucket will have its packets drop precedence set to 2 and its DSCP set to 37. This command only creates a policy, it must be applied to one or more ports to be operational.

Syntax: [no] policy-map *map-name* **cir** *cir-rate* **cbs** *cbs-rate* {**eir** *eir-rate* **ebs** *ebs-rate* **excess-priority** *priority-num* [**excess-dscp** *dscp-num*] | **eir** *eir-rate* **ebs** *ebs-rate* **excess-dp** *dp-val* [**excess-dscp** *dscp-num*]}

The *map-name* variable is the name you will use to reference the policy map in traffic policing command. It can be a character string up to 64 characters long.

The **cir** parameter defines the value of the Committed Information Rate (CIR) as the rate defined in the *cir-rate* variable. Acceptable values are: 0 - 10000000000 bits per second (bps) in increments of 8,144 bits per second (bps).

The **cbs** parameter defines the value of the Committed Burst Size (CBS) as the rate defined in the *cbs-rate* variable. Acceptable values are: 1250 - 1250000000 bytes in increments of 1 byte.

The **eir** parameter defines the value of the Excess Information Rate (EIR) as the rate defined in the *eir-rate* variable. Acceptable values are: 0 - 10000000000 bits per second (bps) in increments of 8,144 bits per second (bps).

The **ebs** parameter defines the value of the Excess Burst Size (EBS) as the rate defined in the *ebs-rate* variable. Acceptable values are: 1250 - 1250000000 bytes in increments of 1 byte.

The **excess-priority** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket will have its packets priority queue set to the value set in the *priority-num* variable. Acceptable values for the *priority-num* are 0-7.

The **excess-dp** parameter specifies the WRED drop precedence for traffic whose bandwidth requirements exceed what is available in the CIR bucket and is sent to the EIR bucket. Acceptable values for the *dp-val* are 0-3. Packets with a value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped.

The **excess-dp** parameter is compared with the **excess-dscp** parameter in bit [2:1] first, then it is converted.

The **excess-dscp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket will have its packets DSCP priority set to the value set in the *dscp-num* variable. Acceptable values for the *dscp-num* are 0-63. When this parameter is used together with the **excess-dp** parameter, the value set for bits 2:1 (zero-based) in the **excess-dscp** parameter must be equal to the value set for **excess-dp** .

Configuring port-based traffic policing for inbound and outbound ports

Port-based traffic policing limits the rate on an individual inbound or outbound physical port to a specified rate.

To configure port-based traffic policing policy for outbound ports, enter commands such as the following at the interface level.

```
device(config)# interface ethernet 1/1
device(config-if-1/1)# rate-limit out 500000000 250000000
```

The commands configure a traffic policing policy for outbound traffic on port 1/1. The policy limits the average rate of all outbound traffic to 500000000 bits per second (bps) with a maximum burst size of 250000000 bits per second (bps).

To configure port based traffic policing policy through a policy map, enter a command such as the following.

```
device(config)# interface ethernet 1/1
device(config-if-1/1)# rate-limit input policy-map map1
```

The commands configure a traffic policing policy for inbound traffic on port 1/1. The policy references the policy map map1 for rate limiting policy parameters.

The complete syntax for configuring a port-based traffic policing policy is:

Syntax: [no] rate-limit {**in** | **out**} {*average-rate maximum-burst* | **policy-map** *map-name*}

The **in** parameter applies the policy to traffic on inbound ports.

The **out** parameter applies the policy to traffic on outbound ports.

Only one inbound and one outbound port-based traffic policing policy can be applied to a port.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the lower multiple of 8,144 bits per second (bps). Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for rate limiting the port and VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in Applying traffic policing parameters using a policy map on page 33.

Policy change remarks

When a rate-limit policy is changed an automated remark is generated. The following examples show the remarks displayed for adding or removing a rate-limit policy.

```
device(config-if-e1000-1/12)#no rate-limit input access-group 102 policy-map abc2
Delete Existing Remark Profile to table 2 with used count 0
device(config-if-e1000-1/12)#rate-limit input access-group 102 policy-map abc2
Add New Remark Profile to table 2 with used count 1
```

Configuring a port and priority-based traffic policing policy for inbound and outbound ports

To configure port based traffic policing policy directly, enter a command such as the following.

```
device(config)# interface ethernet 1/1
device(config-if-1/1)# rate-limit input priority q1 500000000 33553920
```

The commands configure a traffic policing policy for inbound traffic on port 1/1. The policy limits the average rate of all inbound traffic to 500000000 bits per second (bps) with a maximum burst size of 33553920 bits per second (bps) for packets with their priority queue set to 1.

Syntax: [no] rate-limit {**input** | **output**} **priority** *queue-num* {*average-ratemaximum-burst* | **policy-map** *map-name*}

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

Only one port-based traffic policing policy can be applied to a port.

The **priority** parameter specifies the internal queue in the *queue-num* variable which is rate limited by this command.

The priority queues for rate limiting internal queue mapping are shown in the following example.

```
device(config-if-e10000-1/1)#rate-limit input priority
  q0                        priority queue 0 (internal priority 0 and 1)
  q1                        priority queue 1 (internal priority 2 and 3)
  q2                        priority queue 2 (internal priority 4 and 5)
  q3                        priority queue 3 (internal priority 6 and 7)
device(config-if-e10000-1/1)#rate-limit input priority
```

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps). Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The **policy-map** parameter specifies the policy map named in the *map-name* variable to be used to provide parameters for rate limiting the port. This command is only used when configuring rate limiting to a port using a policy map as described in Applying traffic policing parameters using a policy map on page 33.

Configuring a VLAN-based traffic policing policy

To configure a port-and-VLAN based traffic policing policy, enter commands such as the following.

```
device(config)# interface ethernet 1/1
device(config-if-1/1)# rate-limit input vlan 10 500000000 33553920
device(config)# interface ethernet 1/2
device(config-if-1/2)# rate-limit output vlan 20 policy-map map1
```

These commands configure two traffic policing policies that limit the average rate of all inbound traffic on port 1/1 with VLAN tag 10 and all outbound traffic on port 1/2 VLAN tag 20. The first policy limits packets with VLAN tag 10 to an average rate of 500000000 bits per second (bps) with a maximum burst size of 33553920 bytes on port 1/1. The second policy limits packets with VLAN tag 20 to values defined in policy map map1. Tagged packets belonging to VLANs other than 10 and 20 and untagged packets are not subject to traffic policing on these ports.

Syntax: [no] rate-limit {**input** | **output**} [**priority** *queue-num*] **vlan-id** *vlan-num* {*average-rate maximum-burst* | **policy-map** *map-name*}

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

The **priority** parameter specifies an 802.1p value in the *queue-num* variable that is used to identify packets that will be rate limited by this command. This parameter is optional.

The **vlan-id** *vlan-number* parameter species the VLAN ID to which the policy applies. You can specify up to 990 priority or 3960 non-priority VLAN-based traffic policing policies on a port.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps). Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The **policy-map** parameter specifies the policy map named in the *map-name* variable to be used to provide parameters for traffic policing the port and VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in Applying traffic policing parameters using a policy map on page 33.

Configuring a VLAN group-based traffic policing policy

A traffic policing policy can be applied to a VLAN group. VLANs that are members of a VLAN group share the specified bandwidth defined in the traffic policing policy applied to that group.

To configure a traffic policing policy for a VLAN group, perform the following tasks.

1. Define the VLANs that you want to place in a traffic policing VLAN group.
2. Define a rate limiting VLAN group. This VLAN group is specific to the traffic policing feature. Enter commands such as the following.

   ```
   device(config)# rl-vlan-group 10
   device(config-vlan-rate-group)# vlan 3 5 to 7 10
   ```

   The commands assign VLANs 3, 5,6, 7, and 10 to traffic policing VLAN group 10.

   Syntax:[no] rl-vlan-group *vlan-group-number*

   Syntax:[no] vlan *vlan-number* [*to-vlan-number*]

   The **rl-vlan-group** command takes you to the VLAN group traffic policing level. Enter the ID of the VLAN group that you want to create or update by entering a value for *vlan-group-number*.

   Use the **vlan** command to assign or remove VLANs to the rate limiting VLAN group. You can enter the individual VLAN IDs or a range of VLAN IDs.
3. Create a policy for the VLAN group and apply it to the interface you want. Enter commands such as the following.

   ```
   device(config)# interface ethernet 1/1
   device(config-if-1/1)# rate-limit input group 10 500000000 33553920
   ```

   These commands configure a traffic policing policy that limits the average rate of all inbound traffic on port 1/1 from vlan group VlanGroupA. This policy limits packets from VlanGroupA to an average rate of 500000000 bits per second (bps) with a maximum burst size of 33553920 bytes on port 1/1. VLAN Group based traffic policing is only available for inbound ports.

   Syntax:[no] rate-limit input group *vlan-group-id* [**priority** *queue-num*] {*average-rate maximum-burst* | **policy-map** *map-name*}

   The **input** parameter applies the policy to traffic on inbound ports.

   The **priority** parameter specifies an 802.1p value in the *queue-num* variable that is used to identify packets that will be traffic policed by this command. This parameter is optional.

The *vlan-group-id* parameter species the VLAN Group ID to which the policy applies.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits. Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The **policy-map** parameter specifies the policy map named in the *map-name* variable to be used to provide parameters for rate limiting the VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in Applying traffic policing parameters using a policy map on page 33.

Configuring a port and ACL-based rate limiting

You can use standard or extended IP ACLs for port-and-ACL-based rate limiting:

- Standard IP ACLs match traffic based on source IP address information.
- Extended ACLs match traffic based on source and destination IP addresses and IP protocol information. Extended ACLs for TCP and UDP protocol must also match on source and destination IP addresses and TCP or UDP protocol information.
- You can bind multiple rate limiting policies to a single port. However, once a matching ACL clause is found for a packet, the device does not evaluate subsequent clauses in that rate limiting ACL and subsequent rate limiting ACLs.
- You can apply an ACL ID to a port-and-ACL-based traffic policing policy even before you define the ACL. The traffic policing policy does not take effect until the ACL is defined.
- It is not necessary to remove an ACL from a port-and-ACL-based rate limiting policy before deleting the ACL.
- Layer 2 ACL rate limiting is supported.
- The include -control feature extends the NetIron Service Defense functionality to include ARP and other control packets under existing ACL-based rate-limit policies. As of the NetIron 5.9.00 release, the ability to rate-limit protocol control packets and ARP packets is being provided on Gen 2 FPGA-based modules.
- Gen 1.1 modules will partially support the include-control option, but is only applicable to protocol message traffic. ARP packets will not be rate-limited on Gen 1.1 modules. The option is not supported on the BR-MLX-10Gx24-DM module.
- The include-control option to rate-limit ARP and protocol control packets is available in L2, Standard, Extended, and IPv6 ACL rate-limit policies.

Port-and-ACL-based traffic policing is supported for traffic on inbound and outbound ports. To configure port-and-ACL-based traffic policing policies, enter commands such as the following.

```
device(config)#access-list 50 permit host 1.1.1.2 include-control
device(config)#access-list 50 deny host 1.1.1.3
device(config)#access-list 60 permit host 2.2.2.3
device(config-if-1/1)# rate-limit input access-group 50 priority q1 500000000 33553920
device(config-if-1/1)# rate-limit input multicast 100000 10000 include-control
shutdown 1 alert 80000 10000
```

These commands first configure access-list groups that contain the ACLs that will be used in the traffic policing policy. Use the **permit** condition for traffic that will be traffic policed. Traffic that match the **deny** condition are not subject to traffic policing.

Next, the commands configure two traffic policing policies on port 1/1. The policies limit the average rate of all inbound IP traffic that match the permit rules of ACLs 50 and 60. The first policy limits the rate of all permitted IP traffic with a priority queue value of q1 from host 1.1.1.2 to an average rate of 500000000 bits per second (bps) with a maximum burst size of 33553920 bytes. Rate of all traffic from host 1.1.1.3 is not subject to rate limiting since it is denied by ACL 50; it is merely forwarded on the port.

The second policy limits the rate of all IP traffic from host 2.2.2.3 to an average rate of 100000000 bits per second (bps) with a maximum burst size of 268431230 bytes.

All IP traffic that does not match ACLs 50 and 60 are not subject to traffic policing.

Syntax: [no] rate-limit {**input** | **output**} [**vrf** *vrf-name*] **access-group** *group-number* [**priority** *queue-num* {*average-rate maximum-burst* | **policy-map** *map-name*}]

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

The **VRF** parameter specifies that the access-group will only apply to traffic within the VRF whose name is specified in the *vrf-name* variable. This feature is only supported on inbound traffic with Layer-3 ACLs.

The **access-group** *group-number* variable specifies the group number to which the ACLs used in the policy belong.

NOTE
An ACL must exist in the configuration before it can take effect in a traffic policing policy.

The **priority** parameter specifies a priority queue value in the *queue-num* variable that is used to identify packets that will be traffic policed by this command. The possible values for this parameter are: q0, q1, q2, or q3. Multiple queues can be specified. This parameter is optional.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps). Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to Applying traffic policing parameters directly to a port on page 32 for more details.

The **policy-map** parameter specifies the policy map named in the *map-name* variable to be used to provide parameters for traffic policing the VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in Applying traffic policing parameters using a policy map on page 33.

Using ACLs for filtering in addition to rate limiting

When you use the ACL-based mode, the permit and deny conditions in an ACL you use in a rate limiting policy work as follows:

- **permit** - The traffic is rate limited according to the other parameters in the rate limiting policy.
- **deny** - The traffic is forwarded instead of dropped, by default.

You can configure the device to drop traffic that is denied by the ACL instead of forwarding the traffic, on an individual port basis.

NOTE
Once you configure an ACL-based rate limiting policy on a port, you cannot configure a regular (traffic filtering) ACL on the same port. To filter traffic, you must enable the strict ACL option.

To configure the device to drop traffic that is denied by a rate limiting ACL, enter the following command at the configuration level for the port.

```
device(config-if-1/1)# rate-limit strict-acl
```

Syntax:[no] rate-limit strict-acl

Configuring for no priority-based traffic policing

By default, up to 990 different traffic policing policies can be applied to a single 10 GB Ethernet port. This combined with the 4 priorities utilizes 3960 rate limiting classes. You can configure a system-wide policy so that up to 3960 individual traffic policing policies can be applied to a single 10 GB Ethernet port.

To configure a Brocade device to not allow priority-based traffic policing, enter commands such as the following at the interface level.

```
device(config)# qos-policy
device(qos-policy)# no rate-limit internal-priority-based
```

Syntax: [no] rate-limit internal-priority-based

If this command is implemented, the number of different rate limiting policies that can be applied to a single port is increased from 990 to 3960.

Configuring rate limiting for Copied-CPU-bound traffic

A new feature was added that allows you to limit the rate of Copied-CPU-bound packets from applications such as sFlow, ACL logging, RPF logging, and source MAC address learning (with known destination address). This feature can be configured as described in this section

The following command assigns a rate limit of 200,000,000 bits per second (bps) and a priority queue of 0 to copied-CPU-bound incoming traffic on PPCR 1 though its assignment on port 3/2.

```
device(config)# rl-cpu-copy 0 200000000 ethernet 3/2
```

Syntax: rl-cpu-copy *priority-number limit-rate* **ethernet** *slot/port* [**to ethernet** *slot/port*]

The *priority-number* variable specifies the CPU-bound traffic priority queue to apply the rate limiting. This can be a value from 0 to 7.

The *limit-rate* variable specifies the limiting rate for the specified CPU-bound traffic priority queue. Acceptable values are from 1 to 300000000 bits per second (bps). The default rate for all is 300,000,000 bps.

The*slot/port* variable specifies the port that you want to apply copied-CPU-bound rate limiting to. You can apply the command to a range of ports using the **to ethernet** *slot/port* option. When you assign a port, the command applies to all ports that are associated with the same packet processor (PPCR).

TABLE 3 Ports per packet processor

Interface module	Ports Per Packet Processor (PPCR)	
	PPCR1	PPCR2
4 X 10 Gbps	1 - 2	3 - 4
20 X 1 Gbps	1 - 20	

You can display the **rl-cpu-copy** configuration displayed previously using the following command.

```
device# show rl-cpu-copy
```

```
Rate shaping configuration on CPU Copy priority queues
priority 0 200000000 ethernet 3/1 to 3/20
```

Notice that although the command was only executed for port 3/2, it applies to all the ports attached to the same PPCR. In this case ports 3/1 to 3/20.

Syntax: show rl-cpu-copy

Configuring VLL endpoint over an FDP or CDP enabled interface

Configuring VLL endpoint over an FDP/CDP enabled interface will implicitly disable the FDP/CDP configuration and also will be enabled back implicitly when the VLL endpoint is deleted on that specific interface, considering the DP/CDP is enabled globally.

Information messages will be displayed to notify the user as below in these cases:

For example, when VLL endpoint is created, the information messages are as below.

When only FDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
info- FDP is disabled on port 4/3
info- FDP is disabled on port 4/5
info- FDP is disabled on port 4/7
```

When only CDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
info- FDP is disabled on port 4/3
info- FDP is disabled on port 4/5
info- FDP is disabled on port 4/7
```

When both FDP/CDP are enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
info- FDP is disabled on port 4/3
info- FDP is disabled on port 4/5
info- FDP is disabled on port 4/7
```

For example, when the VLL endpoint is deleted the information messages are displayed as below.

When only FDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan -100)# no tag eth 4/3 eth 4/5 eth 4/7
info - FDP is enabled on port 4/3
info - FDP is enabled on port 4/5
info - FDP is enabled on port 4/7
```

When only CDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# no tag eth 4/3 eth 4/5 eth 4/7
info - FDP is enabled on port 4/3
info - FDP is enabled on port 4/5
info - FDP is enabled on port 4/7
```

When both FDP/CDP are enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# no tag eth 4/3 eth 4/5 eth 4/7
info - FDP/CDP is enabled on port 4/3
info - FDP/CDP is enabled on port 4/5
info - FDP/CDP is enabled on port 4/7
```

NOTE
If the VLL endpoint is configured over a globally enabled FDP/CDP interface, the show run command does not display the FDP/CDP information for that specific interface.

NOTE
By removing FDP/CDP from the configuration, the `no fdp enable` and `no cdp enable` stays in the configuration of the VPLS endpoints, which cannot be removed.

Configuring VPLS endpoint over an FDP or CDP interface

Configuring VPLS endpoint over a FDP/CDP enabled interface will implicitly disable the FDP/CDP configuration on that specific interface for that instance, considering FDP/CDP is enabled globally. In this case, the `show run` command will display the running configuration information as shown below.

The following examples explains the show run output for different instances:

The show run output when the VPLS endpoint is configured over a globally enabled FDP/CDP interface:

```
Brocade(config-mpls-vpls-svlan-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
FDP/CDP is disabled on port 4/3
FDP/CDP is disabled on port 4/5
FDP/CDP is disabled on port 4/                                   i7
```

The show run output when the VPLS endpoint is configured over a globally enabled FDP/CDP interface:

```
Brocade(config-mpls-vpls-svlan-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
FDP/CDP is disabled on port 4/3
FDP/CDP is disabled on port 4/5
FDP/CDP is disabled on port 4/7
```

The show run output when the VPLS output is removed over a globally enabled FDP/CDP interface:

```
FDP/CDP is enabled on port 4/3
FDP/CDP is enabled on port 4/5
FDP/CDP is enabled on port 4/7
```

The show run output when the VPLS endpoint is removed over a globally enabled FDP/CDP interface:

```
FDP/CDP is enabled on port 4/3
FDP/CDP is enabled on port 4/5
FDP/CDP is enabled on port 4/7
```

NOTE
If an VPLS endpoint is configured over a globally enabled FDP/CDP interface, the show run will not display FDP/CDP information for that specific interface until the VPLS endpoint is deleted. On deleting the VPLS endpoints, the previous FDP/CDP configuration is retained over that specific interface and the show run displays the FDP/CDP information again for that interface.

NOTE
By removing the FDP/CDP from the configuration, the `no cdp enable` or `no fdp enable` stays in the configuration of the VPLS endpoint, both of which cannot be removed.

Displaying rate limiting policies

Use one of the following commands to view the rate limiting policies that have been configured:

- **show rate counters** - Displays accounting information for rate limit input and output usage.
- **show rate-limit counters** - Displays accounting information for rate limit usage.

- **show rate-limit group** - Displays the VLANs that are in the specified group.
- **show rate-limit** - Displays rate limiting policies implemented per interface.
- **show rate-limit ipv6 hoplimit-expired-to-cpu** - Displays counters for rate-limit configuration, only applicable for GEN-2 and newer modules.
- **show policy map** - Displays rate limiting policies implemented in the configured policy maps.

You can configure a Brocade device to exclude the 20-byte per-packet Ethernet overhead from Traffic Policing byte accounting. This can be done by configuring the **vlan-counter exclude-overhead** command.

Displaying accounting information for rate limit usage

To display accounting information for rate limit input and output usage, enter the following command.

```
device# show rate counters
```

Syntax: show rate counters

Output such as the following will display.

```
device# show rate counters
interface e 4/1
rate-limit output access-group 100 64476048 100000
  Committed Bytes Fwd:        1865392         Drop:  867731400 bytes
  Excess(re-marked) Pkt Fwd:  1864800         Total: 871461592 bytes

interface e 4/1
rate-limit input access-group 101 64476048 100000
  Committed Bytes Fwd:        2755181         Drop:  956733220 bytes
  Excess(re-marked) Pkt Fwd:  5264810         Total: 491663702 bytes

interface e 1/17
 rate-limit input vlan-id 1500 244320 1200000
  Committed Bytes Fwd:     620631             Drop:  1663745 bytes
  Excess(re-marked) Pkt Fwd: 233648           Total: 3852045 bytes
```

Syntax: show rate-limit counters [**interface** *slot/port*]

The *interface slot/port* option allows you to get accounting information for a specified interface only.

Output such as the following will display.

```
device# show rate-limit counters
interface e 2/1
 rate-limit input access-group 400 999993616 1000000000
  Fwd:        0                       Drop:  0 bytes
  Re-mark:    0                       Total: 0 bytes
```

Syntax: show rate-limit counters ipv6 hoplimit-expired-to-cpu

Output such as the following will display.

NOTE
Counters for rate-limit config ("hoplimit-expired-to-cpu") are only applicable for GEN-2 and later modules

```
device# show rate-limit ipv6 hoplimit-expired-to-cpu

Note: Counters for rate-limit config ("hoplimit-expired-to-cpu") are only applicable
for GEN-2 and later modules;
  Committed Bytes Fwd:       38962063104             Drop:  0 bytes
  Excess(re-marked) Pkt Fwd: 0                       Total: 38962063104 bytes
```

TABLE 4 Rate limit counters parameters

This field...	Displays...
Interface	The interface that rate limit accounting information is being displayed for.
rate-limit input	A rate limit configuration that defines rate limit policy for inbound traffic on the defined interface.
rate-limit output	A rate limit configuration that defines rate limit policy for outbound traffic on the defined interface.
Committed Bytes Fwd	The traffic in bytes that has been forwarded from this interface as a result of matching the CIR and CBS rate limit policy value since the device was started up or the counter has been reset.
Fwd	The traffic in bytes that has been forwarded from this interface as a result of this rate limit policy since the device was started up or the counter has been reset.
Drop	The traffic in bytes that has been dropped from this interface as a result of the defined rate limit policy since the device was started up or the counter has been reset.
Excess(re-mark) Pkt Fwd	The number of packets that have been forwarded as a result of exceeding the bandwidth available in the CIR bucket and matching the EIR policy setting. and excess packets whose priority has been changed if re-marking is applied by the rate limit policy.
Total	The total traffic in bytes that has been carried on this interface for the defined rate limit policy since the device was started up or the counter has been reset.

Monitoring rate limit usage by SNMP

Accounting information for rate limit usage can also be monitored by SNMP. The agAclAccntTable supports the following objects, which map to the rate limit usage counters.

agAclAccntRaclDropCnt: drop counters

agAclAccntRaclFwdCnt: fwd counters

agAclAccntRaclRemarkCnt: re-mark counters

agAclAccntRaclTotalCnt: total counters

For detailed information, please refer to the "Filtering Traffic" chapter of the *Unified IP MIB Reference* .

Among other applications, this accounting feature allows per-port VLAN statistics in the inbound or outbound direction to be extracted by means of SNMP. This can be achieved by adding ACL filters for the monitored VLAN on the appropriate port. This accounting feature works for all modules of the Brocade NetIron XMR and Brocade NetIron MLX Series platforms. For modules that do not support extended VLAN statistics, this feature provides a means of extracting per-port VLAN statistics.

Resetting the rate limit counters

You can reset all of the rate limit counters using the following command.

```
device# clear rate-limit counters
```

Syntax: clear rate-limit counters [*interface*]

The *interface* variable specifies an interface that you want to clear the rate limit counters for. If you do not specify an interface, all rate limit counters on the device will be reset.

Displaying information about rate limit VLAN groups

To display information about rate limit VLAN groups, enter the following command.

```
device# show rate-limit group
```

Syntax:show rate-limit group

Output such as the following will display

```
rl-vlan-group 1
 vlan 10 to 15
```

This display shows the following information.

TABLE 5 Rate limit VLAN group parameters

This field...	Displays...
rl-vlan-group	The VLAN group whose contents are displayed.
vlan	VLANs contained in the VLAN group specified.

Displaying rate limit policies per interface

To display information about rate limit policies that are configured per interface, enter the following command.

```
device# show rate-limit
```

Syntax: show rate-limit

Output such as the following will display.

```
device(config-if-e10000-1/1)#show rate-limit
interface e 1/1
 rate-limit input 959904 2000000
 rate-limit output 2986368 2000000
```

This display shows the following information.

TABLE 6 Rate limit interface parameters

This field...	Displays...
rate-limit input	The average-rate and maximum burst rate configured for inbound traffic on the specified interface.

TABLE 6 Rate limit interface parameters (Continued)

This field...	Displays...
rate-limit output	The average-rate and maximum burst rate configured for outbound traffic on the specified interface.

Displaying rate limit policies configured in policy maps

To display information about rate limit policy maps, enter the following command.

```
device# show policy-map
```

Syntax: show policy-map [*map-name*]

The *map-name* variable limits the display of policy map configuration information to the map specified. If this variable is not used, configuration information will be displayed for all policy maps configured on the device.

Output such as the following will display.

```
device(config-policymap pmap1)#show policy-map
policy-map pmap1
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 20000        bytes
  excess-priority 2 excess-dscp 43
policy-map pmap2
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 30000        bytes
  excess-priority 1 excess-dscp 30
```

This display shows the following information.

TABLE 7 Rate limit policy map parameters

This field...	Displays...
policy-map	The name of the policy map whose configuration is being displayed
cir	The value of the Committed Information Rate (CIR) configured for this policy map. Possible values are: 1 - 10000000000 bps.
cbs	The value of the Committed Burst Size (CBS) configured for this policy map. Possible values are: 1250 - 1250000000 bytes.
eir	The value of the Excess Information Rate (EIR) configured for this policy map. Possible values are: 1 - 10000000000 bps.
ebs	The value of the Excess Burst Size (EBS) configured for this policy map. Possible values are: 1250 - 1250000000 bytes.
excess-priority	The priority queue that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket is set to. Possible values are 0-3.
excess-dscp	The priority queue that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket is set to. Possible values are 0-63.

IPv6 ACL-based rate limiting

Rate-limiting is supported for both inbound and outbound traffic on an interface. Rate-limiting IPv6 traffic on an interface is performed by classifying traffic using an IPv6 access-list.

IPv6 ACL based rate-limiting configuration considerations

- IPv6 ACL based rate-limiting can be configured only on a physical interface.
- IPv6 ACL based rate-limiting can be configured separately for inbound and outbound traffic on an interface.
- Multiple IPv6 ACL based rate-limiting policies can be applied to a single port.
- Once a matching ACL clause is hit, subsequent rules and subsequent rate-limiting bindings on the interface are not evaluated.
- An undefined ACL can be used in a rate-limiting configuration.
- When “force-delete-bound-acl” is enabled, an ACL can be deleted even if in use by a rate-limiting policy.
- Whenever the rules of an ACL used in a rate-limiting binding is modified, the changes are not reflected immediately. You must execute the IPv6 ACL rebind command for the changes to take effect.
- IPv4 and IPv6 ACL based rate-limiting configurations can co-exist on an interface.

IPv6 ACL based rate-limiting command options

The following sections discuss the configuration sequence and commands in detail. The following is the entire command syntax. Each of the following configuration commands provide a detailed description of the specific command.

Syntax: [no] rate-limit { **input** [**vrf** *vrf_name*] | **output** } **access-group** { *acl_id* | *name* { **ipv6** | **ipv4** | **mac** } *acl_name* } [{ **priority** *priority_queue* } { *average_rate_bps* | *max_burst_bps* } | { **policy-map** *policy_map_name* }] | { **strict-acl** } | **include-control**

NOTE
The keywords IPv4 and mac are used to configure rate-limiting using named IPv4 and named L2 ACLs. The keyword include-control is used to apply rate-limiting to ARP and other control packets.

IPv6 ACL based rate-limiting configuration sequence

The configuration sequence to configure rate limiting using IPv6 access list include the following.

1. Create the IPv6 access list.
2. Create a policy map.
3. Configure rate limiting on an interface for inbound/outbound traffic using the IPv6 access-list.
    a. Configure the average and maximum burst rate-limit parameters.
    b. Use the policy-map to apply the rate-limit parameters.

Create IPv6 access-list (ACL)

IPv6 access-lists are named access-lists. The following example is an access-list that blocks all Telnet traffic received from IPv6 host 2000:2382:e0bb::2.

```
device(config)# ipv6 access-list fdry
device(config-ipv6-access-list-fdry)# deny tcp host 2000:2382:e0bb::2 any eq
telnet
device(config-ipv6-access-list-fdry)# permit ipv6 any any
device(config-ipv6-access-list-fdry)# exit
```

Create policy-map

The following example configures the traffic policing policy-map map5 to limit CIR rate to 1000000 the CBS rate to 2000000, the EIR rate to 1000000 and the EBS to 2000000.

```
device(config)# policy-map map5
device(config-policymap map5)# cir 1000000 cbs 2000000 eir 1000000 ebs 2000000
excess-dp 2 excess-dscp 37
```

Configure inbound rate-limiting on an interface

This section provides inbound rate-limiting configuration on an interface.

Configure average and maximum burst sizes

The command configures rate-limiting for inbound traffic on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
device(config-if-1/1)# rate-limit input access-group name ipv6 fdry 1000000
2000000
```

Syntax: [no] rate-limit { { **input** [**vrf** *vrf_name*] } **access-group** { *name* { **ipv6** } *acl_name* } [*average_rate_bps* | *max_burst_bps*] include-control

Configure using policy-map

The command configures rate-limiting for inbound traffic on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
device(config-if-1/1)# rate-limit input access-group name ipv6 fdry policy-map map5
include-control
```

include-control

Syntax: [no] rate-limit { { **input** [**vrf** *vrf_name*] } **access-group** { *name* { **ipv6** } *acl_name* } { **policy-map** *policy_map_name* } **include-control**

Configure for a specific priority-queue

The command configures rate-limiting for inbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
device(config-if-1/1)# rate-limit input access-group name ipv6 fdry priority q0
1000000 2000000
```

Syntax: [no] rate-limit { { **input** [**vrf** *vrf_name*] } **access-group** { *name* { **ipv6** } *acl_name* } [{ **priority** *priority_queue* }] { *average_rate_bps* | *max_burst_bps* } **include-control**

The command configures rate-limiting for inbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
device(config-if-1/1)# rate-limit input access-group name ipv6 fdry priority q0
policy-map map5
```

Configure VRF specific rate-limit

IPv6 access-list based rate-limiting can be configured for a specific VRF. Rate-limiting is applied to the inbound traffic for the interfaces which are part of the configured VRF. The following command configures rate-limiting for inbound traffic on the VRF “data” using the access-list “fdry”.

```
device(config-if-1/1)# rate-limit input vrf data access-group name ipv6 fdry
1000000 2000000 include-control
```

Syntax: [no] rate-limit { { **input** [**vrf** *vrf_name*] } **access-group** { *name* { **ipv6** } *acl_name* } { *average_rate_bps* | *max_burst_bps* } **include-control**

NOTE
This feature is not supported in Brocade NetIron CES and Brocade NetIron CER.

Configure outbound rate-limiting on an interface

Use the following steps to configure outbound rate-limiting on an interface.

Configure average and maximum burst sizes

The command configures rate-limiting for outbound traffic on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
device(config-if-1/1)# rate-limit output access-group name ipv6 fdry 1000000
2000000
```

Syntax: [no] rate-limit output access-group {*acl_id* | *name* { **ipv6** } *acl_name*} { *average_rate_bps* | *max_burst_bps* }

Configure using policy-map

The command configures rate-limiting for outbound traffic on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
device(config-if-1/1)# rate-limit output access-group name ipv6 fdry policy-map
map5
```

Syntax: [no] rate-limit output access-group { *name* { **ipv6** } *acl_name* } { **policy-map** *policy_map_name* }

Configure for a specific priority-queue

The command configures rate-limiting for outbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
device(config-if-1/1)# rate-limit output access-group name ipv6 fdry priority q0
1000000 2000000
```

Syntax: [no] rate-limit output access-group { *name* { **ipv6** } *acl_name* } [{ **priority** *priority_queue* }] { *average_rate_bps* | *max_burst_bps* }

The command configures rate-limiting for outbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
device(config-if-1/1)# rate-limit output access-group name ipv6 fdry priority q0
policy-map map5
```

Configure strict-ACL rate-limiting on the interface

By default, rate-limiting is applied to traffic that matches a permit clause. If the traffic does not match any clause or if the traffic matches a deny clause, it is forwarded normally (neither dropped nor rate-limited). You can choose to drop packets that do not match any clause or that match the deny clause by configuring the strict ACL option under an interface.

NOTE
The strict ACL option is independent of ACL type (Layer 2/IPv4/IPv6).

When Strict ACL is enabled without any option, it applies to Layer-2, IPv4, and IPv6 ACL based rate-limiting configured on that port. The following command enables strict-ACL rate-limiting on an interface.

```
device(config-if-1/1)# rate-limit strict-acl
```

Syntax: [no] rate-limit strict-acl

The following IPv6 ACL v6_permit_h2 has a permit and a deny clause.

```
device(config)# ipv6 access-list v6_permit_h2
device(config-ipv6-access-list-v6_permit_h2)# permit ipv6 host 1000:2382:e0bb::1
any
device(config-ipv6-access-list-v6_permit_h2)# deny ipv6 host 3000::1 any
```

The following configuration enables strict ACL option on interface 2/1.

```
device(config)# interface ethernet 2/1
device(config-if-e1000-2/1)# rate-limit strict-acl
device(config-if-e1000-2/1)# rate-limit input access-group name ipv6
v6_permit_h2 policy-map 1mbps
```

Traffic matching the permit clause will be rate-limited as per the rate values of the policy-map 1mbps.

Traffic which does not match any clause or that matches the deny clause will be dropped.

Deleting an IPv6 Access-List which is bound to rate-limit

When user attempts to delete an access-list which is bound to rate-limit profile, an error is thrown to the user that the ACL is in use.

```
device (config)# no ipv6 access-list sample_v6
IPv6 ACL sample_v6 attached to an interface : error - ACL In Use.
```

To delete an IPv6 access-list which is bound to rate-limit profile, use the following configuration.

```
device (config)# acl-policy
device (config-acl-policy)#force-delete-bound-acl
```

After the force-delete-bound-acl configuration is enabled, you can delete any ACL even if it is bound.

Configuring rate-limit using non-existing access-list

Rate-limiting can be configured using a non-existing or empty IPv6 access-list. When the access-list is created or when filters are added to the access-list and an explicit rebind is performed, the rate-limit parameters will be programmed on the interface.

Output of show commands to verify output

Following is the output of the debug commands to confirm the configuration and functionality.

Display Rate-Limit Configuration

```
device# show rate-limit
interface e 1/1
rate-limit input access-group name ipv6 fdry 993568 2000000 include-control
```

Display CAM contents for IPv6 access-list

```
device#show cam v6acl 4/2
LP Index Src IP Addr        SPort IFL/VLAN ID
         Dst IP Addr        DPort Pro Age Out IF PRAM
4 b4000 ::/0                0     N/A
        ::/0                0     0   Dis Pass  000a8
4 b4008 ::/0                34816 N/A
        ::/0                0     58  Dis Pass  000a9
4 b4010 ::/0                34560 N/A
        ::/0                0     58  Dis Pass  000aa
4 b4018 ::/0                0     N/A
        ::/0                0     0   Dis Drop  000ab
```

Display rate-limit configurations

The following show command displays the rate limit configuration on an interface.

Syntax: show rate-limit interface *slot/port* input | output*access-group* { **access-group** { *acl_id* | *name* { **ipv6** | **ipv4** | **mac** } *ACL_NAME*

Display rate-limit counters

The following show command displays rate-limit counters on an interface.

Syntax: show rate-limit counters interface *slot/port* input | output*access-group* { **access-group** { *acl_id* | *name* { **ipv6** | **ipv4** | **mac** } *ACL_NAME*

Display access-list accounting for rate-limiting

The following show command displays rate-limit accounting for an IPv6 ACL.

```
device(config)# show ipv6 access-list accounting eth 1/1 in rate-limit
```

Syntax: show ipv6 access-list accounting *slot/port* **input** | **outputin rate-limit**

Clearing rate-limit counters

You can clear rate-limit counters using the following command:

```
device# clear rate-limit counters ipv6-subnet
```

Syntax: clear rate-limit counters ipv6-subnet

Layer 2 ACL-based rate limiting

Layer 2 ACL-based rate limiting enables devices to limit the rate of incoming traffic in hardware, without CPU intervention. Rate limiting in hardware enables the device to manage bandwidth at line-rate speed.

In general, Layer 2 ACL-based rate limiting works along the same lines as hardware-based rate limiting feature. All the rules and regulations that apply to hardware-based rate limiting also apply to this feature.

Configuration rules and notes

- You can apply Layer 2 ACL-based rate limiting on a physical port. You cannot apply it to a virtual interface or a LAG port.
- You cannot use IPv4 ACL-based filtering and Layer 2 ACL-based rate limiting on the same port. You can, however, configure one port on the device to use IP ACLs and another port on the same device to use Layer 2 ACL-based rate limiting.
- You cannot use IPv4 ACL-based rate limiting and Layer 2 ACL-based rate limiting on the same port. You can, however, configure one port on the device to use IPv4 ACL-base rate limiting and another port on the same device to use Layer 2 ACL-based rate limiting.
- You can bind multiple rate limiting policies to a single port. However, once a matching ACL clause is found for a packet, the device does not evaluate subsequent clauses in that rate limiting ACL and subsequent rate limiting ACLs.
- Only number ACLs support rate limiting
- Layer 2 rate limiting ACLs will function with vlan-cpu-protection, broadcast and multicast limiting features. If incoming traffic matches an inbound Layer 2 rate limiting ACL, it is first rate-limited based on the policy. If packets are not dropped due to rate limiting, they are forwarded either to the CPU or flooded in the VLAN according to the vlan-cpu-protection feature.

NOTE
The above behavior applies only for the Brocade NetIron XMR and Brocade NetIron MLX Series, not for the Brocade NetIron CES and Brocade NetIron CER. For the Brocade NetIron CES and Brocade NetIron CER platforms, once the broadcast and multicast limiting features are enabled, these limits will take precedence over the defined port rate-limit.

- The broadcast and multicast packet limiting feature limits packets in the CPU, while the Layer 2 ACL RL is a network processing (NP) RL feature. Packets are first subjected to the Layer 2 ACL RL at the NP. Once packets are forwarded to CPU, the broadcast and multicast limiting feature begins functioning and packets may be dropped in the CPU if the rate exceeds the limit.

Editing a Layer 2 ACL Table

You can make changes to the Layer 2 ACL table definitions without unbinding and rebinding the rate limit policy. For example, you can add a new clause to the ACL table, delete a clause from the table, or delete the ACL table that is used by a rate limit policy.

Define rate limiting parameters

To define rate limiting parameters, enter commands such as the following:

```
device(config)#policy-map map1
device(config-policymap map1)#cir 1000000 cbs 2000000 eir 1000000 ebs 2000000 excess-
dp 2
```

Binding Layer 2 ACL-based rate limiting policy to a port

To bind an Layer 2 ACL based rate-limiting policy on a specific port, enter commands such as the following:

```
device(config-policymap map1)#int eth 14/1
device(config-if-e10000-14/1)# rate-limit input access-group 400 policy-map map1
```

Syntax: [no] rate-limit {**input** | **output**} **access-group** *num* **policy-map** *map-name*

Specifying rate limiting parameters without a policy map

To specify rate-limiting without using a policy map, enter a command such as the following:

```
device(config-if-e10000-14/1)# rate-limit input access-group 400 49999998416
75000000000
```

Syntax: [no] rate-limit {**input** | **output**} **access-group** *acl-id average-rate maximum-burst*

The *acl-id* for Layer 2 ACLs can range from 400 to 499.

The *average-rate* is the maximum number of bits the policy allows during one second.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have.

This command is only used when configuring traffic policing directly to a port as described in Applying traffic policing parameters directly to a port on page 32.

Display accounting

To display access list accounting, enter a command such as the following.

```
device#show access-list accounting eth 14/1 in rate-limit
Collecting L2 ACL accounting for 400 on port 14/1  ...  Completed successfully.
RL ACL Accounting Information:
Inbound: ACL 400
    0:   permit 0000.0000.0021 ffff.ffff.ffff any any etype any
       Hit count: (1 sec)                   0   (1 min)                 0
                  (5 min)                   0   (accum)                 0
```

Rate limiting protocol traffic using Layer 2 inbound ACLs

Using interface level Layer 2 inbound ACLs, you can rate limit the following types of protocol traffic by explicitly configuring a filter to match the traffic:

- STP/RSTP/BPDU
- MRP
- VSRP
- LACP
- GARP
- UDLP

To rate-limit all such control traffic enter commands such as the following:

```
device(config)#access-list 402 permit any 0180.c200.0000 ffff.ffff.ffff any etype any
device(config)#access-list 402 permit any 0304.8000.0000 ffff.ffff.ffff any etype any
device(config)#access-list 402 permit any 0304.8000.0100 ffff.ffff.ff00 any etype any
device(config)#access-list 402 permit any 0180.c200.0002 ffff.ffff.ffff any etype any
device(config)#access-list 402 permit any 0180.c200.0020 ffff.ffff.fff0 any etype any
device(config)#access-list 402 permit any 00e0.5200.0000 ffff.ffff.ffff any etype any
device(config)#access-list 402 deny any any any etype any
```

TABLE 8 Filters for protocols

Protocol	Filter
STP/RSTP/BPDU	access-list 402 permit any 0180.c200.0000 ffff.ffff.ffff any etype any
MRP	access-list 402 permit any 0304.8000.0000 ffff.ffff.ffff any etype any
VSRP	access-list 402 permit any 0304.8000.0100 ffff.ffff.ff00 any etype any
LACP	access-list 402 permit any 0180.c200.0002 ffff.ffff.ffff any etype any
GARP	access-list 402 permit any 0180.c200.0020 ffff.ffff.fff0 any etype any
UDLP	access-list 402 permit any 00e0.5200.0000 ffff.ffff.ffff any etype any

NOTE
The filters must have the specific destination MAC address as shown above in the configuration. You can filter all protocols as shown in the previous configuration example above, or only specific protocols.

Example of Layer 2 ACL to rate limit broadcast traffic

To define an ACL that rate limits broadcast traffic and forwards all other traffic without rate limiting, enter commands such the following:

```
device(config)#access-list 411 permit any ffff.ffff.ffff ffff.ffff.ffff
device(config)#access-list 411 deny any any
```

To bind an ACL that rate limits broadcast traffic and forwards all other traffic without rate limiting, enter commands such the following:

```
device(config)#int eth 14/1
device(config-if-e10000-14/1)#rate-limit in access-gr 411 8144 100
```

Rate limiting ARP packets

You can limit the rate of ARP traffic that requires CPU processing on Brocade devices, such as ARP request traffic, and ARP response addressed to the device. The feature is set globally and applies to all ARP traffic received at the device. With this feature you can apply a defined policy map to all ARP traffic bound for the CPU.

When the **vlan-cpu-protection** command is configured, ARP request packets are switched within a VLAN by the hardware and thus cannot be rate-limited by the **ip rate limit arp policy-map** command. To limit the rate of ARP packets that are forwarded by hardware, use interface-level, layer-2 inbound ACLs with the "etype arp" option.

Configuring rate limiting of ARP packets

To rate limit ARP packets bound for the CPU using a policy map named "limitarp", enter the following command.

```
device(config)# ip rate-limit arp policy-map limitarp
```

Syntax: [no] ip rate-limit arp policy-map *map-name*

The *map-name* variable is the name of the policy map to be used to provide parameters for rate limiting CPU-bound ARP packets. If the policy map specified has not been defined, the rate limit values are initialized to the line rate values.

Displaying statistics for ARP rate limiting

You can display ARP Rate Limiting Statistics using the following command.

```
device# show rate-limit arp
 Fwd:        1865392                Drop:  867731400 bytes
 Re-mark:    1864800                Total: 871461592 bytes
```

Syntax: show rate-limit arp

This display shows the following information.

TABLE 9 Rate limit ARP display parameters

Parameter	Description
Fwd	The ARP traffic in bytes that has been sent to the CPU as a result of the ARP rate limit policy since the device was started up or the counter was reset.
Drop	The ARP traffic in bytes that has been dropped as a result of the ARP rate limit policy since the device was started up or the counter was reset.
Re-mark	The ARP traffic in bytes whose priority have been remarked as a result of exceed the bandwidth available in the CIR bucket for the ARP rate limit policy since the device was started up or the counter was reset.
Total	The total ARP traffic in bytes that has been subjected to the ARP rate limit policy since the device was started up or the counter was reset.

Clearing Statistics for ARP Rate Limiting

You can clear ARP Rate Limiting Statistics using the following command:

```
device# clear rate-limit arp
```

Syntax: clear rate-limit arp

Rate-Limiting of CPU bound IP options and TTL<=1 packets

Configuring rate-limiting for IP option and TTL<=1 packets

Overview:

CPU assistance is required to properly support the IPv4 packets with options header or IPv4/IPv6 packets with (ttl<=1). Handling of these packets in the software is an extra overhead for CPU since primary task of CPU is to handle protocol packets and other management related tasks of the Network device. This behavior can be easily used to attack the CPU on the device with flood of options or ttl<=1 packets. Hence to safeguard the CPU from such DOS attacks, this feature gives an option to rate-limit such CPU bound packets.

NOTE
Configuring new TTL rate-limits and 990 VLAN rate-limits simultaneously is not supported. The rate-limit which is configured later will over-write the Rx (inbound) RL policy, the Tx policy is not affected.

Configuring rate-limit policy for IP option and TTL<=1 packets.

The following commands configure rate-limit policy for IP option and TTL<=1 packets.

Configuration steps:

1. Create a rate-limit policy-map using policy-map command.
2. Provision rate-limit values in policy-map.
3. Apply the policy-map for IP option and TTL<=1 packets.

Configuration example

Creating and provisioning a rate-limit policy-map.

The following commands create and provisions a CPU bound rate-limit policy-map.

The example below explains how to create CPU bound rate-limit policy map.

```
Brocade(config)# policy-map save-cpu-policy
Brocade(config-policymap save-cpu-policy)# cir 1000000 cbs 2000000 eir 1000000 ebs 
2000000
Brocade(config-policymap save-cpu-policy)# exit
```

Configuring rate-limit option on IPv4 ttl-expired-to-cpu packets.

The following command configures rate-limit option on IPv4 ttl-expired-to-cpu packets.

```
Brocade(config)#ip rate-limit ttl-expired-to-cpu policy-map save-cpu-policy
```

Syntax: [no] ip rate-limit ttl-expired-to-cpu policy-map *policy-map-name*

NOTE

The following warning message is displayed if only some of the cards are supported and few are not supported. For example cards 1 and 3 do not support the feature

WARNING: rate-limit config for protocol "**ttl-expired-to-cpu**" is not supported on module 1, 3

The following warning message is displayed if none of the cards are supported.

WARNING: rate-limit config for protocol "**ttl-expired-to-cpu**" is not supported on available modules. It is only supported on GEN-2 and later modules.

Configuring rate-limit option on IPv6 hoplimit-expired-to-cpu packets.

The following command configures rate-limit option on IPv6 hoplimit-expired-to-cpu packets.

```
Brocade(config)#ipv6 rate-limit hoplimit-expired-to-cpu policy-map save-cpu-policy
```

Syntax: [no] ipv6 rate-limit hoplimit-expired-to-cpu policy-map *rate-limit policy*

NOTE

The following warning message is displayed if only some of the cards are supported and few are not supported.

WARNING: rate-limit config for protocol "**hoplimit-expired-to-cpu**" is not supported on module 1, 3

The following warning message is displayed if none of the cards are supported.

WARNING: rate-limit config for protocol "**hoplimit-expired-to-cpu**" is not supported on available modules. It is only supported on GEN-2 and later modules.

Configuring rate-limit on IPv4 option-pkt-to-cpu packets.

The following command configures rate-limit on IPv4 option-pkt-to-cpu packets.

```
Brocade(config)#ip rate-limit option-pkt-to-cpu policy-map save-cpu-policy
```

Syntax: [no] ip rate-limit option-pkt-to-cpu policy-map *rate-limit policy*

NOTE

The following warning message is displayed if only some of the cards are supported and few are not supported.

WARNING: rate-limit config for protocol "**option-pkt-to-cpu**" is not supported on module 1, 3

The following warning message is displayed if none of the cards are supported.

WARNING: rate-limit config for protocol "**option-pkt-to-cpu**" is not supported on available modules. It is only supported on GEN-2 and later modules.

NOTE

The ttl-expired-to-cpu rate-limiting has the higher precedence than option-pkt-to-cpu rate-limiting.

ip rate-limit ttl-expired-to-cpu policy-map

Applies rate-limit option on IPv4 ttl packets, if the ttl count is less than or equal to one.

Syntax **[no] ip rate-limit ttl-expired-to-cpu policy-map** *rate-limit policy*

Command Default By default this command is disabled.

Parameters *rate-limit policy*

Name of the policy-map.

Modes Global Configuration Mode.

Usage Guidelines The **no** form of the command disables rate-limit option on IPv4 ttl-expired-to-cpu packets.

Create CPU bound rate-limit policy map before applying rate-limiting for ttl-expired-to-cpu packets.

NOTE

The following warning message is displayed if only some of the cards are supported and few are not supported.

WARNING: rate-limit config for protocol "**ttl-expired-to-cpu**" is not supported on module 1, 3

The following warning message is displayed if none of the cards are supported.

WARNING: rate-limit config for protocol "**ttl-expired-to-cpu**" is not supported on available modules. It is only supported on GEN-2 and later modules.

Examples The following example explains how to apply rate-limit option on IPv4 ttl-expired-to-cpu packets.

```
Brocade(config)#ip rate-limit ttl-expired-to-cpu policy-map save-cpu-policy
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

ip rate-limit option-pkt-to-cpu policy-map

Applies rate-limit on IPv4 option packets.

Syntax **[no] ip rate-limit option-pkt-to-cpu policy-map** *rate-limit policy*

Command Default By default this command is disabled.

Parameters *rate-limit policy*

Name of the policy-map.

Modes Global Configuration Mode.

Usage Guidelines The **no** form of the command disables rate-limiting on IPv4 option packets.

Create CPU bound rate-limit policy map before applying rate-limiting for option packets.

NOTE

The following warning message is displayed if only some of the cards are supported and few are not supported.

WARNING: rate-limit config for protocol "**option-pkt-to-cpu**" is not supported on module 1, 3

The following warning message is displayed if none of the cards are supported.

WARNING: rate-limit config for protocol "**option-pkt-to-cpu**" is not supported on available modules. It is only supported on GEN-2 and later modules.

Examples The following example explains how to apply rate-limit for IPv4 option packets.

```
Brocade(config)#ip rate-limit option-pkt-to-cpu policy-map save-cpu-policy
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

ipv6 rate-limit hoplimit-expired-to-cpu

Applies rate-limit option on IPv6 hop-limit packets, if the hop-limit count is less than or equal to one.

Syntax

ipv6 rate-limit hoplimit-expired-to-cpu *rate-limit policy*

no ipv6 rate-limit hoplimit-expired-to-cpu *rate-limit policy*

Command Default

By default no rate-limit option is applied to IPv6 hop-limit packets, if the hop-limit count is less than or equal to one.

Parameters

rate-limit policy

Name of the policy-map.

Modes

Global configuration mode

Usage Guidelines

The **no** form of the command disables rate-limit option on IPv6 hop-limit packets.

Create CPU bound rate-limit policy map before applying rate-limiting for hop-limit packets.

NOTE

The following warning message is displayed if only some of the cards are supported and few are not supported.

```
Warning: rate-limit config for protocol "hoplimit-expired-to-cpu" is not supported on
module 1, 3
```

NOTE

The following warning message is displayed if none of the cards are supported.

```
Warning: rate-limit config for protocol "hoplimit-expired-to-cpu" is not supported on
available modules.
It is only supported on GEN-2 and later modules.
```

Examples

The following example explains how to apply a rate-limit policy for IPv6 hop-limit packets.

```
device(config)# ipv6 rate-limit hoplimit-expired-to-cpu policy-map save-cpu-policy
```

History

Release version	Command history
5.8.00	This command was introduced.

show rate-limit ttl-expired-to-cpu

Displays the information about rate-limit configuration on IPv4 ttl-expired-to-cpu packets.

Syntax **show rate-limit ttl-expired-to-cpu**

Modes This command operates in all mode.

Command Output The **show rate-limit ttl-expired-to-cpu** command displays the following information:

Output field	Description
Fwd	The ttl-expired-to-cpu traffic in bytes that has been sent to the CPU as a result of the ttl-expired-to-cpu rate limit policy since the device was started up or the counter was reset.
Drop	The ttl-expired-to-cpu traffic in bytes that has been dropped as a result of the ttl-expired-to-cpu rate limit policy since the device was started up or the counter was reset.
Re-mark	The ttl-expired-to-cpu traffic in bytes whose priority have been remarked as a result of exceed the bandwidth available in the CIR bucket for the ttl-expired-to-cpu rate limit policy since the device was started up or the counter was reset.
Total	The total ttl-expired-to-cpu traffic in bytes that has been subjected to the ttl-expired-to-cpu rate limit policy since the device was started up or the counter was reset.

Examples Display output of the **show rate-limit ttl-expired-to-cpu** command:

```
Brocade# show rate-limit ttl-expired-to-cpu
Fwd: 1865392 Drop: 867731400 bytes
Re-mark: 1864800 Total: 871461592 bytes
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

show rate-limit option-pkt-to-cpu

Displays the information about rate-limit configuration on IPv4 option packets.

Syntax **show rate-limit option-pkt-to-cpu**

Modes This command operates in all mode.

Command Output The **show rate-limit option-pkt-to-cpu** command displays the following information:

Output field	Description
Fwd	The IPv4 option-pkt-to-cpu traffic in bytes that has been sent to the CPU as a result of the IPv4 option-pkt-to-cpu rate limit policy since the device was started up or the counter was reset.
Drop	The IPv4 option-pkt-to-cpu traffic in bytes that has been dropped as a result of the IPv4 option-pkt-to-cpu rate limit policy since the device was started up or the counter was reset.
Re-mark	The IPv4 option-pkt-to-cpu traffic in bytes whose priority have been remarked as a result of exceed the bandwidth available in the CIR bucket for the IPv4 option-pkt-to-cpu rate limit policy since the device was started up or the counter was reset.
Total	The total IPv4 option-pkt-to-cpu traffic in bytes that has been subjected to the IPv4 option-pkt-to-cpu rate limit policy since the device was started up or the counter was reset.

Examples Display output of the **show rate-limit option-pkt-to-cpu** command:

```
Brocade# show rate-limit option-pkt-to-cpu
Fwd: 1865392 Drop: 867731400 bytes
Re-mark: 1864800 Total: 871461592 bytes
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

show rate-limit ipv6 hoplimit-expired-to-cpu

Displays the information about rate-limit configuration on IPv6 hoplimit-not-ok packets.

Syntax **show rate-limit ipv6 hoplimit-expired-to-cpu**

Modes This command operates in all mode.

Command Output The **show rate-limit ipv6 hoplimit-expired-to-cpu** command displays the following information:

Output field	Description
Fwd	The hoplimit-expired-to-cpu traffic in bytes that has been sent to the CPU as a result of the hoplimit-expired-to-cpu rate limit policy since the device was started up or the counter was reset.
Drop	The hoplimit-expired-to-cpu traffic in bytes that has been dropped as a result of the hoplimit-expired-to-cpu rate limit policy since the device was started up or the counter was reset.
Re-mark	The hoplimit-expired-to-cpu traffic in bytes whose priority have been remarked as a result of exceed the bandwidth available in the CIR bucket for the hoplimit-expired-to-cpu rate limit policy since the device was started up or the counter was reset.
Total	The total hoplimit-expired-to-cpu traffic in bytes that has been subjected to the hoplimit-expired-to-cpu rate limit policy since the device was started up or the counter was reset.

Examples Display output of the **show rate-limit ipv6 hoplimit-expired-to-cpu** command:

```
Brocade#show rate-limit ipv6 hoplimit-expired-to-cpu
Fwd: 1865392 Drop: 867731400 bytes
Re-mark: 1864800 Total: 871461592 bytes
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

clear rate-limit counters ip-ttl-expired-to-cpu

Clears the rate-limit counters for IPv4 ttl-expired-to-cpu packets.

Syntax **clear rate-limit counters ip-ttl-expired-to-cpu**

Modes This command operates in all mode.

Examples The following example shows how to clear the rate-limit counters for ip-ttl-expired-to-cpu.

```
Brocade# clear rate-limit counters ip-ttl-expired-to-cpu
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

clear rate-limit counters ip-option-pkt-to-cpu

Clears the rate-limit counters for IPv4 option packets.

Syntax **clear rate-limit counters ip-option-pkt-to-cpu**

Modes This command operates in all mode.

Examples The following example shows how to clear the rate-limit counters for IPv4 option packets.

```
Brocade# clear rate-limit counters ip-option-pkt-to-cpu
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

clear rate-limit counters ipv6-hoplimit-expired-to-cpu

Clears the rate-limit counters for IPv6 hoplimit-expired-to-cpu packets.

Syntax **clear rate-limit counters ipv6-hoplimit-expired-to-cpu**

Modes This command operates in all mode.

Examples The following example shows how to clear the rate-limit counters for hoplimit-expired-to-cpu packets.

```
Brocade# clear rate-limit counters ipv6-hoplimit-expired-to-cpu
```

History

Release version	Command history
Multi-Service IronWare Release 5.8.00	This command was introduced.

Configuring Traffic Policing for the Brocade NetIron CES and Brocade NetIron CER

Traffic policing on Brocade NetIron CES and Brocade NetIron CER devices

Brocade NetIron CES and Brocade NetIron CER devices provide line-rate traffic policing in hardware on inbound and outbound ports.

You can configure a device to use one of the following traffic policing modes:

- **Port-based** - Limits the rate on an individual physical port to a specified number. Only one inbound and one outbound port-based traffic policing policy can be applied to a port. (Refer to Configuring port-based traffic policing for inbound and outbound ports on page 73.) These policies can be applied to inbound and outbound traffic.
- **Port-and-ACL-based** - Limits the rate of IP traffic on an individual physical port that matches the permit conditions in IP Access Control Lists (ACLs). Layer-2 ACL-based traffic policing is supported. You can use standard or extended IP ACLs. Standard IP ACLs match traffic based on source IP address information. Extended ACLs match traffic based on source and destination IP address and IP protocol information. Extended ACLs for TCP and UDP also match on source and destination TCP or UDP addresses, and protocol information. These policies can be applied to inbound and outbound traffic. Brocade NetIron CES and Brocade NetIron CER devices support up to 3967 (1984 egress and 1983 ingress) policies for a port per packet processor (PPCR).

Multi-Service IronWare software lets you apply traffic policing parameters directly to a port, or create a policy map to define a set of traffic policing parameters and apply that policy map to one or more ports.

Applying traffic policing parameters directly to a port

When you apply a traffic policing parameters directly to a port, two parameters are specified: average rate and maximum burst. These parameters configure credits and credit totals.

Average rate

The average rate is the maximum number of bits a port can receive during a one-second interval. The rate of the traffic will not exceed the average rate as specified by the traffic policing policy.

The average rate represents a percentage of line rate (bandwidth) for an interface, expressed in bits per second (bps). It cannot be larger than the line rate for the port. For the Brocade NetIron CES and Brocade NetIron CER devices, the average rate can be entered in as any value from 0 up to the line rate of the port.

Maximum burst

Maximum burst allows a higher-than-average rate to traffic that meets the rate limiting criteria. Traffic is allowed to pass through the port for a short period of time. The unused bandwidth can be accumulated up to a maximum equal to the maximum burst value.

Credits and credit total

Each rate limiting policy is assigned a class. The class uses the average rate and maximum burst in the rate limit policy to calculate credits and credit totals.

Credit size is measured in bytes. A credit is a forwarding allowance for a traffic policed port, and is the smallest number of bytes that can be allowed during a rate limiting interval. Minimum credit size can be 1 byte.

During a rate limiting interval, a port can send or receive only as many bytes as the port has credits for. For example, if an inbound rate limiting policy results in a port receiving two credits per rate limiting interval, the port can send or receive a maximum of 2 bytes of data during that interval.

In each interval, the number of bytes equal to the credit size is added to the running total of the class. The running total of a class represents the number of bytes that can pass without being subject to rate limiting.

The second parameter is the maximum credit total . The maximum credit total is based on the maximum burst value and is measured in bytes.

The running total can never exceed the maximum credit total. When a packet arrives at the port, a class is assigned to the packet based on the traffic policing policies. If the running total of the class is less than the size of the packet, the packet is dropped. Otherwise, the size of the packet is subtracted from the running total and the packet is forwarded. If there is no traffic that matches traffic policing criteria, then the running total can grow up to the maximum credit total.

Applying traffic policing parameters using a policy map

The policy map configuration ties a policy name to a set of traffic policing policies. The policy name is then applied to ports that you want to rate limit using the defined policy. This allows you to set a policy in a single location the affects multiple ports and to make changes to that policy. Refer to Configuring traffic policing on page 71.

In the policy map configuration, the traffic policing policy determines the rate of inbound or outbound traffic (in bits per second or bps) that is allowed per port. This traffic is initially traffic policed by a Committed Information Rate (CIR) bucket. Traffic that is not accommodated in the CIR bucket is then subject to the Excess Information Rate (EIR) bucket.

The CIR bucket

The CIR rate limiting bucket is defined by two parameters: the CIR rate, and the Committed Burst Size (CBS) rate. The CIR rate is the maximum number of bits a port is allowed to receive or send during a one-second interval. The rate of the traffic that matches the traffic policing policy can not exceed the CIR rate. The CIR rate represents a portion of the line rate (bandwidth) for an interface expressed in bits per second (bps) and cannot be larger than the line rate of the port. CIR-defined traffic that does not use the available CIR rate accumulates credits that be used later in circumstances where it temporarily exceeds the CIR rate.

When traffic exceeds the bandwidth that has been reserved for it by the CIR rate defined in its policy, it becomes subject to the CBS rate. The CBS rate is higher than the CIR rate to traffic that exceeds the CIR rate. The bandwidth in the CBS rate accumulates during periods when traffic that has been defined by a policy does not use the full CIR rate available. Traffic is allowed to pass through the port for a short period of time at the CBS rate.

When inbound or outbound traffic exceeds the bandwidth available for the defined CIR and CBS rates, it is either dropped, or made subject to the conditions set in the EIR bucket.

The EIR bucket

The EIR bucket provides an option for traffic that has exceeded the conditions set by policy for the CIR bucket. In the EIR bucket, two parameters define traffic that is available: the Excess Information Rate (EIR) and the Excess Burst Size (EBS) rate. The EIR and EBS operate exactly like the CIR and CBS except that they only act upon traffic that has been passed to the EIR bucket because it could not be accommodated by the CIR bucket. Like the CIR, the EIR provides an initial bandwidth allocation to accommodate inbound and outbound traffic. If the bandwidth provided by the EIR is insufficient to accommodate the excess traffic, the defined EBS rate provides for burst traffic. Like the CBS, the bandwidth available for burst traffic from the EBS is subject to the amount of bandwidth that is accumulated during periods of time when traffic that has been allocated by the EIR policy is not used.

In addition to providing additional bandwidth for traffic that exceeds that available for the CIR bucket, traffic rate limited by the EIR bucket can have its excess priority and excess dscp values changed. Using this option, priority parameters are set following the EBS value that change the priority of traffic that is being rate limited using the EIR bucket.

Configuration considerations

- Only one type of traffic policing policy can be applied on a physical port. For example, you cannot apply port-and-ACL-based and port-based traffic policing policies on the same port.
- The maximum burst in a traffic policing policy cannot be less than the average rate and cannot be more than the port line rate.
- Control packets are not subject to traffic policing.
- Source MAC address with Virtual Leased Line (VLL) endpoints are not subject to traffic policing.
- Up to four different sets of excess parameters are supported. Delete or unbind other policy maps from other interfaces. You can also change the excess parameters in the policy map to match one of the existing profiles to share the Remark Profile Tables.
- BUM rate-limit may not work properly with a lower configured-rate and bursty traffic.
- IPv4 ACL-based rate limiting is not supported on VPLS and VLL endpoints.
- IPv4 ACL-based rate limiting on a port that belongs to a VLAN is not supported on a VLAN without a VE configured.
- IPv4 ACL-based rate limiting is not supported on a port that belongs to a VLAN where in Layer3 Interface(VE) is configured for MPLS.

Limitations

In the Brocade NetIron CES and Brocade NetIron CES, UDP rate-limiting is applicable only in the following scenarios:

- When sending 1% of 1G traffic with packet size of 64 bytes to the device for configured Burst-max value (up to 8000)
- When sending 10% of 1G traffic with packet size of 64 bytes to the device for configured Burst-max value (up to 1500)
- When sending 100% of 1G traffic with packet size of 64 bytes to the device for configured Burst-max value (up to 500).

Configuring traffic policing

The following sections show examples of how to configure each type of traffic policing.

Configuring a policy map

To configure a policy map, enter commands such as these.

```
device(config)#policy-map map1
device(config-policymap map1)#cir 1000000 cbs 500000 eir 1000000 ebs 500000
```

This command configures traffic policing policy-map map1 to limit CIR rate to 1000000, the CBS rate to 500000, the EIR rate to 1000000, and the EBS to 500000. This command only creates a policy, it must be applied to one or more ports to be operational.

Syntax: [no] policy-map *map-name*

Syntax: [no] cir *cir-rate* **cbs** *cbs-rate* **eir** *eir-rate* **ebs** *ebs-rate* [**excess-dp** *dp-val* **excess-dscp** *dscp-num* **excess-priority** *priority-num* **excess-pcp** *pcp-num* **excess-exp** *exp-num*]

The *map-name* variable is the name you use to reference the policy map in traffic policing command. It can be a character string up to 64 characters long.

The **cir** parameter defines the value of the CIR as the rate defined in the *cir-rate* variable. Acceptable values are: 0 - 100,000,000,000 bps.

The **cbs** parameter defines the value of the CBS as the rate defined in the *cbs-rate* variable. Acceptable values are: 1250 - 12,500,000,000 bytes in increments of 1 byte.

The **eir** parameter defines the value of the EIR as the rate defined in the *eir-rate* variable. Acceptable values are: 0 - 100,000,000,000 bps.

The **ebs** parameter defines the value of the EBS as the rate defined in the *ebs-rate* variable. Acceptable values are: 1250 - 12,500,000,000 bytes in increments of 1 byte.

The following parameters are optional. If configured, they will specify the remarking of the Quality of Service parameters, if the rate is over the limits that are specified by CIR or CBS, but less than the limit of EIR and EBS.

The **excess-dp** parameter specifies the drop precedence for traffic whose bandwidth requirements exceed what is available in the CIR bucket and is sent to the EIR bucket. Acceptable values for the *dp-val* are 0 - 3. Packets with a value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped.

For Drop Precedence, the Brocade NetIron MLX Series and Brocade NetIron XMR has 4 levels, while Brocade NetIron CER and Brocade NetIron CES have 3 levels. The Brocade NetIron CER and Brocade NetIron CES internally convert the 4 levels as follows:

0 -> 0, 1 -> 1, 2 -> 1, 3 -> 2

NOTE
excess-dp is not supported on egress.

The **excess-dscp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will have their DSCP priority set to the value set in the *dscp-num* variable. Acceptable values for the *dscp-num* are 0 - 63. When this parameter is used together with the **excess-dp** parameter, the value set for bits 2:1 (zero-based) in the **excess-dscp** parameter must be equal to the value set for excess-dp. The **excess-dp** parameter is compared with **excess-dscp** in bit [2:1] first, then it is converted.

The **excess-priority** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will have their priority queue or TC set to the value set in the *priority-num* variable. Acceptable values for the *priority-num* are 0 - 7.

The **excess-pcp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will have their pcp or UP (user priority) set to the value set in the *pcp-num* variable. Acceptable values for the *pcp-num* are 0 - 7.

The **excess-exp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will their EXP set to the value set in the *exp-num* variable. Acceptable values for the *exp-num* are 0-7.

Configuring a policy-map to remark profile tables

The Brocade NetIron CER and CES devices and use four remark profile tables per packet processor. The profile determines the CoS remapping tables sets to use, and each profile contains the 5 excess parameters (i.e., excess-dp, excess-dscp, excess-priority, excess-pcp, excess-exp), which remark the Quality of Service parameters if the rate is over the limits that are specified in the CIR, CBS, EIR and EBS parameters.

The profile is selected per port configuration. For the Ingress policier, the profile is selected per the source port; and for the Egress policier, the profile is selected per target port. You can configure as many policy-maps as you want. However, when you apply or bind the policy-map to an interface, only four different profiles are available, where each of the four profiles may be applied to multiple ports.

Note that a policy-map includes both rates and sizes and the 5 excess parameters. A remark profile table has the 5 excess parameters only, so multiple policy-maps may share a single remark profile table, if the 5 excess parameters match.

Configuring port-based traffic policing for inbound and outbound ports

Port-based traffic policing limits the rate on an individual inbound or outbound physical port to a specified rate.

To configure port-based traffic policing policy for outbound ports, enter commands such as the following at the interface level.

```
device(config)# interface ethernet 1/1
device(config-if-1/1)# rate-limit out 500000000 750000000
```

These commands configure a traffic policing policy for outbound traffic on port 1/1. The policy limits the average rate of all outbound traffic to 500 Mbps with a maximum burst size of 750 MBps.

Configuring port-based traffic policing using a policy map

To configure port based traffic policing policy through a policy map, enter commands such as the following.

```
device(config)# interface ethernet 1/1
device(config-if-1/1)# rate-limit input policy-map map1
```

These commands configure a traffic policing policy for inbound traffic on port 1/1. The policy references policy map1 for rate limiting policy parameters.

The complete syntax for configuring a port-based traffic policing policy is:

Syntax: [no] rate-limit {**in** | **out**} {*average-rate maximum-burst* | **policy-map** *map-name*}

The **in** parameter applies the policy to traffic on inbound ports.

The **out** parameter applies the policy to traffic on outbound ports.

Only one inbound and one outbound port-based traffic policing policy can be applied to a port.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. For the Brocade NetIron CER and CES devices, the Average Rate can be entered in as any value from 0 up to the line rate of the port. Refer to Applying traffic policing parameters directly to a port on page 69 for more details.

The *maximum-burst* parameter specifies the extra bits above the average rate that traffic can have. Refer to Applying traffic policing parameters directly to a port on page 69 for more details.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for rate limiting the port and VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in Applying traffic policing parameters using a policy map on page 70.

Configuring a port-and-ACL-based traffic policing policy

You can use standard or extended IP ACLs for port-and-ACL-based traffic policing:

- Standard IP ACLs match traffic based on source IP address information.
- Extended ACLs match traffic based on source and destination IP addresses and IP protocol information. Extended ACLs for TCP and UDP protocols must also match on source and destination IP addresses and TCP or UDP protocol information.
- You can apply an ACL ID to a port-and-ACL-based traffic policing policy before you define the ACL. The traffic policing policy does not take effect until the ACL is defined.
- It is not necessary to remove an ACL from a port-and-ACL-based rate limiting policy before deleting the ACL.
- Layer-2 ACL rate limiting is supported.
- IPv4 ACL-based rate limiting is not supported on VPLS and VLL endpoints.
- IPv4 ACL-based rate limiting on a port that belongs to a VLAN is not supported on a VLAN without a VE configured.
- IPv4 ACL-based rate limiting is not supported on a port that belongs to a VLAN where in Layer3 Interface(VE) is configured for MPLS.

Port-and-ACL-based traffic policing is supported for traffic on inbound and outbound ports. To configure port-and-ACL-based traffic policing policies, enter commands such as the following.

```
device(config)#access-list 50 permit host 1.1.1.2
device(config)#access-list 50 deny host 1.1.1.3
device(config)#access-list 60 permit host 2.2.2.3
device(config-if-1/1)# rate-limit input access-group 50 500000000 20480
device(config-if-1/1)# rate-limit input access-group 60 100000000 24194240
```

These commands first configure access-list groups that contain the ACLs that will be used in the traffic policing policy. Use the **permit** condition for traffic that will be policed. Traffic that matches the **deny** condition is not subject to traffic policing.

Next, the commands configure two traffic policing policies on port 1/1. The policies limit the average rate of all inbound IP traffic that matches the permit rules of ACLs 50 and 60. The first policy limits the rate of all permitted IP traffic from host 1.1.1.2 to an average rate of 500 Mbps with a maximum burst size of 20480 Mbits. Traffic from host 1.1.1.3 is not subject to rate limiting since it is denied by ACL 50; it is merely forwarded on the port.

The second policy limits the rate of all IP traffic from host 2.2.2.3 to an average of 100 Mbps with a maximum burst size of 4194240 Mbits.

Traffic that does not match ACLs 50 and 60 is not subject to traffic policing.

Syntax: no rate-limit {**input** | **output**} **access-group** *group-number* {*average-rate maximum-burst* | **policy-map** *map-name*}

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

The **access-group** *group-number* variable specifies the group number to which the ACLs used in the policy belong.

NOTE
An ACL must exist in the configuration before it can take effect in a traffic policing policy.

The *average-rate* variable specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bps. Refer to Applying traffic policing parameters directly to a port on page 69 for more details.

The *maximum-burst* variable specifies the extra Mbits above the average rate that traffic can have. Refer to Applying traffic policing parameters directly to a port on page 69 for more details.

The **policy-map** parameter specifies the policy map named in the *map-name* variable to be used to provide parameters for traffic policing the VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in Applying traffic policing parameters using a policy map on page 70.

Configuring VLL endpoint over an FDP or CDP enabled interface

Configuring VLL endpoint over an FDP/CDP enabled interface will implicitly disable the FDP/CDP configuration and also will be enabled back implicitly when the VLL endpoint is deleted on that specific interface, considering the DP/CDP is enabled globally.

Information messages will be displayed to notify the user as below in these cases:

For example, when VLL endpoint is created, the information messages are as below.

When only FDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
info- FDP is disabled on port 4/3
info- FDP is disabled on port 4/5
info- FDP is disabled on port 4/7
```

When only CDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
info- FDP is disabled on port 4/3
info- FDP is disabled on port 4/5
info- FDP is disabled on port 4/7
```

When both FDP/CDP are enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
info- FDP is disabled on port 4/3
info- FDP is disabled on port 4/5
info- FDP is disabled on port 4/7
```

For example, when the VLL endpoint is deleted the information messages are displayed as below.

When only FDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan -100)# no tag eth 4/3 eth 4/5 eth 4/7
info - FDP is enabled on port 4/3
info - FDP is enabled on port 4/5
info - FDP is enabled on port 4/7
```

When only CDP is enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# no tag eth 4/3 eth 4/5 eth 4/7
info - FDP is enabled on port 4/3
info - FDP is enabled on port 4/5
info - FDP is enabled on port 4/7
```

When both FDP/CDP are enabled globally.

```
device(config-mpls-vll-vll1-vlan-100)# no tag eth 4/3 eth 4/5 eth 4/7
info - FDP/CDP is enabled on port 4/3
info - FDP/CDP is enabled on port 4/5
info - FDP/CDP is enabled on port 4/7
```

NOTE
If the VLL endpoint is configured over a globally enabled FDP/CDP interface, the show run command does not display the FDP/CDP information for that specific interface.

NOTE
By removing FDP/CDP from the configuration, the `no fdp enable` and `no cdp enable` stays in the configuration of the VPLS endpoints, which cannot be removed.

Configuring VPLS endpoint over an FDP or CDP interface

Configuring VPLS endpoint over a FDP/CDP enabled interface will implicitly disable the FDP/CDP configuration on that specific interface for that instance, considering FDP/CDP is enabled globally. In this case, the `show run` command will display the running configuration information as shown below.

The following examples explains the show run output for different instances:

The show run output when the VPLS endpoint is configured over a globally enabled FDP/CDP interface:

```
Brocade(config-mpls-vpls-svlan-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
FDP/CDP is disabled on port 4/3
FDP/CDP is disabled on port 4/5
FDP/CDP is disabled on port 4/                                  i7
```

The show run output when the VPLS endpoint is configured over a globally enabled FDP/CDP interface:

```
Brocade(config-mpls-vpls-svlan-vlan-100)# tag eth 4/3 eth 4/5 eth 4/7
FDP/CDP is disabled on port 4/3
FDP/CDP is disabled on port 4/5
FDP/CDP is disabled on port 4/7
```

The show run output when the VPLS output is removed over a globally enabled FDP/CDP interface:

```
FDP/CDP is enabled on port 4/3
FDP/CDP is enabled on port 4/5
FDP/CDP is enabled on port 4/7
```

The show run output when the VPLS endpoint is removed over a globally enabled FDP/CDP interface:

```
FDP/CDP is enabled on port 4/3
FDP/CDP is enabled on port 4/5
FDP/CDP is enabled on port 4/7
```

NOTE
If an VPLS endpoint is configured over a globally enabled FDP/CDP interface, the show run will not display FDP/CDP information for that specific interface until the VPLS endpoint is deleted. On deleting

the VPLS endpoints, the previous FDP/CDP configuration is retained over that specific interface and the show run displays the FDP/CDP information again for that interface.

NOTE
By removing the FDP/CDP from the configuration, the `no cdp enable` or `no fdp enable` stays in the configuration of the VPLS endpoint, both of which cannot be removed.

Using ACLs for filtering in addition to rate limiting

When you use the ACL-based mode, the permit and deny conditions in an ACL you use in a rate limiting policy work as follows:

- **Permit** - The traffic is rate limited according to the other parameters in the rate limiting policy.
- **Deny** - The traffic is forwarded instead of dropped, by default.

You can configure the device to drop traffic that is denied by the ACL instead of forwarding the traffic, on an individual port basis.

NOTE
Once you configure an ACL-based rate limiting policy on a port, you cannot configure a regular (traffic filtering) ACL on the same port. To filter traffic, you must enable the strict ACL option.

To configure the device to drop traffic that is denied by a rate limiting ACL, enter the following command at the configuration level for the port.

```
device(config-if-1/1)# rate-limit strict-acl
```

Syntax:[no] rate-limit strict-acl

Displaying rate limiting policies

Use one of the following commands to view the rate limiting policies that have been configured:

- **show rate limit counters** - Displays accounting information for rate limit usage. Only ACL based counters are displayed.
- **show rate limit** - Displays rate limiting policies implemented per interface.
- **show policy map** - Displays rate limiting policies implemented in the configured policy maps.

You can configure a device to exclude the 20-byte per-packet Ethernet overhead from traffic policing byte accounting using the **vlan-counter exclude-overhead** command.

Displaying accounting information for rate limit usage

To display accounting information for rate limit usage, enter the following command.

```
device# show rate-limit counters
```

Syntax: show rate-limit counters *slot/port*

The *slot/port* option allows you to get accounting information for a specified interface only.

Output such as the following is displayed.

```
device# show rate-limit counters
interface e 1/1
 rate-limit input 959904 2000000
  Fwd:        10000                  Drop:  1000 bytes
  Re-mark:    0                      Total: 11000 bytes
 rate-limit output 2986368 2000000
  Fwd:        20000                  Drop:  2340 bytes
  Re-mark:    0                      Total: 22340 bytes
This display shows the following information.
```

TABLE 10 Rate limit counters parameters

This field...	Displays...
rate-limit input	Defines rate limit policy for inbound traffic on the defined interface.
rate-limit output	Defines rate limit policy for outbound traffic on the defined interface.
Fwd	Traffic (in bytes) that has been forwarded as a result of this rate limit policy since the device was started or the counter was reset.
Drop	Traffic (in bytes) that has been dropped as a result of the defined rate limit policy since the device was started up or the counter has been reset.
Re-mark	The number of packets for which priority has been remarked as a result of exceeding the bandwidth available in the CIR bucket for this rate limit policy.
Total	Total traffic (in bytes) that has been carried on this interface for the defined rate limit policy since the device was started or the counter was reset.

NOTE
Port based rate limit counters are not supported for Brocade NetIron CER and CES devices.

Resetting the rate limit counters

You can reset all of the rate limit counters using the following command.

```
device# clear rate-limit counters
```

Syntax: clear rate-limit counters *slot/port*

The *slot/port* variable specifies a port for which you want to clear the rate limit counters. If you do not specify a port, all rate limit counters on the device are reset.

Displaying rate limit policies for a specific interface

To display information about rate limit policies that are configured for a specific interface, enter the following command at the interface level.

```
device(config-if-e10000-1/1)#show rate-limit
```

Syntax: show rate-limit

Output such as the following is displayed.

```
interface e 1/1
 rate-limit input 959904 2000000
 rate-limit output 2986368 2000000
```

This display shows the following information.

TABLE 11 Rate limit interface parameters

This field...	Displays...
rate-limit input	The average-rate and maximum burst rate configured for inbound traffic on the specified interface.
rate-limit output	The average-rate and maximum burst rate configured for outbound traffic on the specified interface.

Displaying rate limit policies configured in policy maps

To display information about rate limit policy maps, enter the following command.

```
device(config-policymap)#show policy-map pmap1
```

Syntax: show policy-map *name*

The *name* variable limits the display to the map specified. If this variable is not used, configuration information is displayed for all policy maps configured on the device.

Output such as the following is displayed.

```
policy-map pmap1
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 20000        bytes
  excess-priority 2 excess-dscp 43
policy-map pmap2
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 30000        bytes
  excess-priority 1 excess-dscp 30
```

This display shows the following information.

TABLE 12 Rate limit policy map parameters

This field...	Displays...
policy-map	The name of the policy map for which the configuration is being displayed
cir	The value of the CIR configured for this policy map. Possible values are: 1 - 100000000000 bps.
cbs	Value of the CBS configured for this policy map. Possible values are: 1250 - 12500000000 bytes.
eir	Value of the EIR configured for this policy map. Possible values are: 1 - 100000000000 bps.
ebs	Value of the EBS configured for this policy map. Possible values are: 1250 - 12500000000 bytes.

Rate limiting BUM packets

To prevent the CPU from being flooded by the broadcast, unknown-unicast, and multicast (BUM) packets or bytes, you can restrict the number of BUM packets received on a port. If a high rate of BUM traffic is received by the device on a given port, you can configure the per-port ingress rate limit for the BUM traffic. When you configure a BUM rate limit, the device accepts the maximum configured number of packets and drops additional BUM packets. A port can have BUM rate limits independent of its interface module. The BUM packets entering the device are rate limited before being replicated.

When the received BUM traffic exceeds the pre-defined rate limit, you can close or shut down the physical port using the **shutdown** option of the **rate-limit** command. When the port is configured to be shut down and the BUM traffic experiences packet drops due to the BUM rate limit, the port is shut down automatically. The port shut down occurs within 2.5 seconds after the BUM traffic exceeds the defined limit. The port can be enabled again using the **no** form of the **rate-limit** command.

Limitations of the BUM rate limit

The configuration of the BUM rate limit has the following limitations:

- The order of the rate-limit command execution is unknown-unicast first, followed by multicast and then broadcast, as shown below.

```
device(config-if-e10000-1/1)#rate-limit input vlan-id 2 unknown-unicast 100000
100000 include-control shutdown
rate-limit input vlan-id 2 multicast 200000 10000 include-control shutdown
rate-limit input vlan-id 2 broadcast 300000 10000 include-control shutdown
```

- The alternative to the ordered command execution is to configure the BUM rate-limit in a single command, as shown below.

```
device(config-if-e10000-1/1)#rate-limit input vlan-id 2 unknown-unicast
100000 100000 includ e-control shutdown
Average rate is adjusted to 97728 bits per second.
device(config-if-e10000-1/1)#rate-limit input vlan-id 2 multicast
200000 10000 include-control shutdown
Average rate is adjusted to 195456 bits per second.
device(config-if-e10000-1/1)#rate-limit input vlan-id 2 broadcast
300000 10000 include-control shutdown
Average rate is adjusted to 293184 bits per second.
```

- The BUM rate limit applies only to ingress ports.
- The per-port rate limit cannot be configured individually per traffic type of the broadcast, unknown unicast, or multicast traffic.
- The port shutdown cannot be configured per traffic type.
- The BUM rate limit does not drop packets based on the priorities.
- When the port reaches the configured rate limit, the device will check if the **shutdown** option is enabled for the port.
- The device counts the rate of BUM packets received on the port, for every port configured for shutdown.
- A single drop counter moves over each port to check for the **shutdown** option in a round robin fashion.
- If the drop counter finds the BUM packets dropped on a port, the port will be shut down until the port is explicitly enabled.
- Global BUM rate limiting is not supported on non-default ESI configured interfaces.
- The rate-limit is configured to count for a minimum of 10 milliseconds (ms) for a 1 GbE port and 1 ms for a 10 GbE port.

- The granularity of the rate limit is 51200 bits for 1 Gigabit per second (Gbps) port and 512000 bits for 10 GbE port.
- Due to hardware limitations, 10G ports with BUM rate-limit can shutdown with a rate lower than the actual configured shutdown threshold rate. When the minimum rate of 512Kbit/sec is configured, 10K bit/sec will be rate limited. With a large shutdown threshold rate 2G bit/sec is configured, 1.5G bit/sec will be rate limited. This limitation does not affect 1G ports.

Configuring per-port rate limiting for BUM traffic

For example, to configure the rate limit on BUM traffic packets to a million bits per second, on port 1/1, enter the following command.

```
device(config-if-e1000-1/1)# rate-limit input broadcast unknown-unicast multicast
1000000 shutdown
```

Syntax: [no] rate-limit input {**broadcast** | **unknown-unicast** | **multicast**} [*average-rate*] [**alert**] [**shutdown**]

The **input** parameter applies the rate limiting policy to traffic on inbound ports.

The **broadcast**, **unknown-unicast** and **multicast** parameters define a rate limit for ingress broadcast, unknown-unicast, and multicast packets on the port. Any combination of these parameters can be used to define the rate limit.

The *average-rate* variable specifies the maximum number of bits a port is allowed to receive during a one-second interval and is the aggregate sum of the broadcast, unknown-unicast, and multicast packets rate limit, if the rate limit is configured for all three packets. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps).

The **alert** option enables an alert when packets are dropped. Another alert is sent when the packet drop has stopped.

The **shutdown** option specifies that the port is to be shut down if the amount of BUM traffic exceeds the pre-defined limit.

When the user tries to add or modify an existing BUM rate limiting policy, the following error message is displayed.

```
Error: There is already a rate limit policy applied on the port.
```

When the BUM traffic exceeds the defined rate limit, port 1/1 is shut down and the reason for the shutdown is displayed in the output of the **show interface** command.

```
device# show interface ethernet 1/1
GigabitEthernet1/1 is down (rate-limit BUM), line protocol is down
STP Root Guard is disabled, STP BPDU Guard is disabled
```

The following syslog message is displayed with the port shutdown information in the output to the **show log** command.

```
device# show log
Nov  4 23:07:52:I:BUM rate-limit is shutting down port 0 on PPCR 0
Nov  4 23:07:52:I:System: Interface ethernet 1/1, state down - shut down by rate-
limiting broadcast, unknown unicast & multicast
```

To enable the shutdown port, delete the previous rate limit by entering the **clear rate-limit bum interface** *slot/port* command.

```
device(config-if-e1000-1/1)# clear rate-limit bum interface 1/1
```

NOTE
If the user binds different types of rate limiting, such as access group, ACL-based, and BUM rate limits to an interface, the lowest rate limit is configured on the interface.

Displaying BUM rate limit information

You can use show commands to display the following information about the BUM rate limits:

- Accounting information for the BUM rate limit
- BUM rate limit policies per interface

Displaying accounting information for the BUM rate limit

To display the accounting information for the BUM rate limit, enter the following command.

```
device# show rate-limit counters bum-drop
interface 1/1 to 1/24                    Drop: 560656640 bytes
interface 1/25 to 1/48                Drop: 212962201728 bytes
interface 2/1 to 2/2                        Drop: 148174664000 bytes
```

Syntax: show rate-limit counters bum-drop

TABLE 13 Output parameters of the show rate-limit counters bum-drop command

Field	Description
interface	Shows the interface for which the accounting information is displayed.
Drop	Shows the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy.

Displaying BUM rate limit policies per interface

To display the BUM rate limit policies that are configured per interface, enter the following command.

```
device# show rate-limit
interface e 1/2
rate-limit input broadcast unknown-unicast multicast 972800
interface e 1/12
rate-limit input broadcast multicast 102400 shutdown
```

Syntax: show rate-limit

TABLE 14 Output parameters of the show rate-limit command

Field	Description
interface	Shows the interface for which the BUM rate limit policy information is displayed.
rate-limit input broadcast unknown-unicast multicast	Shows the average rate configured for the inbound broadcast, unknown-unicast, and multicast traffic on the interface.

TABLE 14 Output parameters of the show rate-limit command (Continued)

Field	Description
rate-limit input broadcast multicast	Shows the average rate and the port shutdown option configured for inbound broadcast and multicast traffic on the interface.

Clearing accounting information for the BUM rate limit

To clear the accounting information for the BUM rate limit, enter the following command.

```
device# clear rate-limit counters bum-drop
```

Syntax: clear rate-limit counters bum-drop

Configuring Quality of Service (QoS) for the Brocade NetIron CES and Brocade NetIron CER Series

Quality of Service (QoS)

The Quality of Service (QoS) features offer many options for the Brocade NetIron CES and Brocade NetIron CER devices.

Quality of Service (QoS) provides preferential treatment to specific traffic, possibly at the expense of other traffic. Without QoS, the Brocade NetIron CES or Brocade NetIron CER device offers best-effort service to each packet and transmits packets without any assurance of reliability, delay bounds, or throughput. Implementing QoS in a network makes performance more predictable and bandwidth utilization more effective.

QoS implementation in the Brocade NetIron CES or Brocade NetIron CER device complies with the IETF-DiffServ and IEEE 802.1p standards. A typical QoS model deployment is based on the following elements:

- At the network edge, the packet is assigned to a QoS service. The service is assigned based on the packet header information (i.e., packet is trusted) or on the ingress interface configuration (packet is not trusted).
- The QoS service defines the packet’s internal QoS handling (e.g., traffic class and drop precedence) and optionally the packet’s external QoS marking, through either the IEEE 802.1p User Priority or the IP header DSCP field.
- Subsequent Brocade NetIron CES and Brocade NetIron CER devices within the network core provide consistent QoS treatment to traffic, based on the packet’s IEEE 802.1p, or DSCP marking. As a result, an end-to-end QoS behavior is provided.
- A Brocade NetIron CES or Brocade NetIron CER device may modify the assigned service if a packet stream exceeds the configured profile. In this case, the packet may be dropped or reassigned to a lower QoS service.
- The Brocade NetIron CES and Brocade NetIron CER devices incorporate the required QoS features to implement network-edge as well as network-core devices.
- The Brocade NetIron CES and Brocade NetIron CER devices provide flexible mechanisms to classify packets into different service levels.
- The packet header may have its User Priority fields set to reflect the QoS assignment.

- Service application mechanism is based on eight egress priority queues per port (including the CPU port), on which congestion-avoidance and congestion-resolution policies are applied.
- QoS encode policies are not supported on the NetIron CES and NetIron CER devices due to Hardware limitations.

QoS model

This chapter describes how QoS is implemented and configured in the Brocade NetIron CES and Brocade NetIron CER devices. The chapter contains the following sections.

Traffic types

Data - Data packets can be either Network-to-Network traffic or traffic from the CPU. Network-to-Network traffic is considered Data traffic.Qos parameters can be assigned and modified for data traffic.

Control - Packets to and from the CPU is considered control traffic. The QoS parameters fro this traffic are preassigned and not configurable.

Setting packet header QoS fields

The device supports setting or modifying the packet header IEEE 802.1p User Priority or IP-DSCP.

Packet QoS attributes

Every packet classified as Data is assigned a set of QoS attributes that can be modified by each ingress pipeline engine.

Each of the ingress pipeline engines contain several Initial QoS Markers that assign the packet's initial QoS attribute.

The ingress pipeline engine also contains a QoS Remarker that can modify the initial QoS attributes.

Even though Brocade NetIron CES and Brocade NetIron CER devices support four drop precedence values 0,1,2 and 3 internally 1 and 2 are assigned the same drop precedence level. The four levels are kept for CLI compatibility with other Brocade devices. Three internal level of drop precedence are 0, {1,2} and 3. in terms of commonly used color based terminology: 0 represents Green (lowest drop precedence}, 1 and 2 represents yellow (higher drop precedence) and 3 represents Red (highest drop precedence).

TABLE 15 Packet QoS attributes

QoS parameter	Description
TC (Traffic Class)	This is the priority level assigned to the packet. When the TxQ enqueues the packet, it uses this field to select the appropriate priority queue.
DP (Drop Precedence)	The TxQ uses this field for congestion resolution. Packets with higher drop precedence are more likely to be discarded in the event of congestion.

Ingress Traffic processing through a device

The QoS operation on Ingress traffic of a Brocade device involves reception and processing of packets based upon priority information contained within the packet. As the packets are processed through the device, there are several opportunities to influence the processing by configuration as described in the following steps.

1. Derive priority and drop precedence from the packets PCP (802.1p) value. The Priority Code Point (PCP) is a 3-bit field within an 802.1Q tagged frame that is used to convey the priority of the frame. By using a mapping table, the 3-bit PCP field can be decoded to derive priority and drop precedence information. Note: the PCP field was formerly called 802.1p.
2. Derive priority and drop precedence from the packets DSCP value.
3. Force the priority and drop precedence value based on the value configured for the physical port.
4. Force the priority value based on an ACL look-up. This is used for setting a a specific priority for and L2, L3 or L4 traffic flow.

NOTE
DEI value will remain 0 regardless of PCP or DSCP value.

Recognizing inbound packet priorities and mapping to internal priority

Internal priority is determined after collecting priority and drop precedence information from various portions of the packet header and determining if a priority value should be forced.

Stage 1 Collect priority and drop precedence information from various portions of the packet header

- If a packet's EtherType matches 8100 or the port's EtherType, derive a priority value and drop precedence by decoding the PCP value.
- For IPv4 packets, derive a priority value and drop precedence by decoding the DSCP bits.
- The derived values for PCP, and DSCP are mapped to a default map.
- To assist the device in the decoding process described in "stage 1", decode-map tables are defined.

Stage 2 Determine if a priority value should be forced

- If a packet's EtherType matches 8100 or the port's EtherType, derive a priority value and drop precedence by decoding the PCP value
- If the **qos pcp force** command is configured on the port, the priority and drop precedence values are set to the value read from the PCP bits.
- If the **qos dscp force** command is configured on the port, the priority and drop precedence values are set to the value read from the DSCP bits.
- If none of the qos force commands are configured, the priority and drop precedence values of a given packet is obtained in the following manner:

  - For Tagged and Untagged IPv4Packets: Priority and drop precedence values obtained from decoded DSCP values.
  - For Tagged Non-IPv4 Packets: Priority and drop precedence values obtained from decoded PCP values.
  - For Untagged, Non-IPv4 Packets: Priority and drop precedence values obtained from priority and drop precedence assigned on a port. If no priority and drop precedence is assigned on a port default value of Priority 0 and Drop Precedence 0 is picked.

Forcing the priority of a packet

Once a packet's ingress priority has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to "force" the priority of a packet based on the following criteria:

- Forced to a priority configured for a specific ingress port. The **priority force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the DSCP priority bits. The **qos dscp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the PCP priority bits. The **qos pcp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is based on an ACL match. The **priority-force** keyword can be used within an ACL to apply a priority to specified traffic.

NOTE
Commands to "force" the priority of a packet are not supported for the **vlan priority** command in the Brocade NetIron CES and Brocade NetIron CER platforms.

If multiple commands containing the **priority-force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

1. ACL match
2. Physical port priority value
3. DSCP value in an incoming IPv4 packet
4. PCP value in a tagged frame or PCP

   Details of how to configure the force commands are provided in Configuring a force priority on page 90.

ACL QoS Considerations

The following should be considered when configuring QoS.

- All packets with ACL match where action is Priority/Priority force will have BOTH UP and EXP bit encoded.
- Regardless of the Port Encoding settings.The effect will be user visible only if the outgoing packet is MPLS for non-MPLS, packets the functionality will remain same as before.
- The EXP bit is carried to the Remote VPLS/VLL in the inner label (VC Label). QoS assignment is on the remote.
- PE device will be based on this EXP. The outgoing packet's PCP value will be encoded based on QoS assigned by the EXP.

- At Ingress:
  - ACL-->TC Assignment ---> TC Encodes {UP, EXP}
- At Remote VPLS:
  - EXP Based QoS Assignment-->TC--> TC Encodes {UP}

Custom decode support

User defined decode maps are supported on the Brocade NetIron CES and Brocade NetIron CER. The custom decode maps have the following implication for QoS handling:

- Per port custom decode maps are not supported. Only a single global QoS map is supported.
- A number of custom decode maps can be defined in the Multi-Service IronWare, but only one can be active at any time in the hardware.
- A user defined map can be applied to replace any existing map including the default map.
- User defined maps are supported for PCP, EXP and DSCP.
- Custom decode maps are supported. This means that packet can be decoded by any one of the many user defined map currently active in the hardware but there will be one and only encode map. Encoding will always be based on default encode map.
- The packet will be decoded as per the decode map, but during encoding NOT all the values will be encoded correctly.
- If the packet is decoded based on DSCP value, the Brocade NetIron CES and Brocade NetIron CER will decode the packet to correct internal priority (Traffic Class) and Drop Precedence. This QoS will be internally respected and packet will be treated in accordance with the assigned Priority and DP. During encoding (if enabled) however, the Brocade NetIron CES and Brocade NetIron CER will encode all the packet QoS attributes as shown below:
  - Assume trust mode is DSCP and encoding policy is turned on for UP and DSCP
  - Assume custom Decode Map: DSCP 56 to Priority 0 and DP 0
  - Encode map is always default
  - IPv4 tagged packet comes with DSCP 56 and UP 5
  - The packet will be assigned Internal Priority of 0 and DP of 0
  - The packet going out will have 802.1p value of 0
  - The packet will NOT have expected DSCP value of 0, instead it will go out unchanged (56)
- If the packet is decoded based on PCP value (which is possible if qos trust mode is PCP force or no trust mode is forced but packet is Tagged and payload is not IPv4), the Brocade NetIron CES will decode the packet to correct internal priority (Traffic Class) and Drop Precedence. This QoS will be internally respected and packet will be treated in accordance with the assigned Priority and DP. During encoding (if enabled) however, Brocade NetIron CES and Brocade NetIron CER will encode all the packet QoS attributes as shown below:
  - Assume trust mode is DSCP and encoding policy is turned on for UP and DSCP
  - Assume custom Decode Map: PCP 5 to Priority 0 and DP 0
  - Encode map is always default
  - Assume an IPv4 tagged packet comes with DSCP 56 and UP 5
  - The packet will be assigned Internal Priority of 0 and DP of 0
  - The packet going out will have DSCP value of 0
  - The packet will not have expected PCP value of 0, instead it will go out unchanged (5)

Forcing the drop precedence of a packet

Once a packet's ingress drop precedence has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to "force" the drop precedence of a packet based on the following criteria:

- Forced to a drop precedence configured for a specific ingress port. The **drop-precedence force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the DSCP priority bits. The **qos dscp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the PCP priority bits. The **qos pcp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is based on an ACL match. The **drop-precedence force** keyword can be used within an ACL to apply a priority to specified traffic.

If multiple commands containing the **force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

1. ACL match
2. Physical port's drop precedence value
3. DSCP value in an incoming IPv4 packet
4. PCP value in a tagged frame or PCP

   Details of how to configure the force commands are provided in Configuring a force priority on page 90.

Configuring QoS

The QoS configuration process involves separate procedures for Ingress and Egress QoS Processing as described in the following major sections.

Configuring QoS procedures applicable to Ingress and Egress

The following procedures are required to configure procedures applicable to both Ingress and Egress QoS Processing on a Brocade device:

- Support for QoS configurations on LAG ports - If you are configuring the enhanced QoS feature on ports within a LAG, refer to Configuring port-level QoS commands on LAG ports on page 92.

Configuring a force priority

In situations where there are conflicting priority values for packets on an Ingress port, that conflict can be resolved by performing a priority merge or by using a **force** command to direct the device to use a particular value above other values. A **force** command can be configured for each of the following:

- Force to the values configured on a port
- Force to the value in the DSCP bits
- Force to the value in the PCP bits
- Force to a value specified within an ACL

Configuring a force priority for a port

You can configure an ingress port with a priority to apply to packets that arrive on it using the **priority** command.

To configure an ingress port with a priority, use the **priority** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)priority 6
```

Syntax: priority *priority-value*

The *priority-value* variable is a value between 0 and 7. The default value is 0.

Once a port has been configured with a priority using the **priority** command, you can then configure the port (using the **priority force** command) to force the configured priority when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the port-configured priority, use the **priority force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)priority force
```

Syntax: [no] priority force

The priority will be forced to zero if no value is configured.

Configuring a force drop precedence for a port

You can configure an ingress port with a drop precedence to apply to packets that arrive on it using the **drop-precedence** command.

To configure an ingress port with a drop precedence, use the **drop-precedence** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)drop-precedence 3
```

Syntax: [no] drop-precedence *dp-value*

The *dp-value* variable is a value between 0 and 3.

Once a port has been configured with a drop precedence using the **drop-precedence** command, you can then configure the port (using the **drop-precedence force** command) to force the configured drop precedence when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the port-configured drop precedence, use the **drop-precedence force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)drop-precedence force
```

Syntax: [no] drop-precedence force

Configuring force priority to the DSCP value

You can configure an ingress port (using the **qos dscp force** command) to force the configured DSCP value when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the DSCP value, use the **qos dscp force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos dscp force
```

Syntax: [no] qos dscp force

Configuring force priority to the PCP value

You can configure an ingress port (using the **qos pcp force** command) to force the configured PCP value when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the PCP value, use the **qos pcp force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos pcp force
```

Syntax: [no] qos pcp force

Configuring force priority to a value specified by an ACL

You can use the **priority-force** keyword within an ACL to apply a priority to specified traffic as described in "Filtering and priority manipulation based on 802.1p priority" in the *Multi-Service IronWare Security Configuration Guide* .

Configuring extended-qos-mode

The **extended-qos-mode** command should only be turned on when deploying a Brocade CES or Brocade CER as MPLS PE devices, if preserving passenger DSCP is required, when terminating VPLS and VLL traffic at the egress end point.

NOTE
You must write this command to memory and perform a system reload for this command to take effect.

NOTE
This command will reduce the hardware table size by half. If the existing configuration has already used more the half of the hardware table size, this command will not succeed. This command will only succeed if there is sufficient space in the hardware table.

You can revert to normal mode using the **no extended-qos-mode** command without losing functionality or behavior.

```
device(config)# extended-qos-mode
```

Syntax: [no] extended-qos-mode

Configuring port-level QoS commands on LAG ports

When applying port-level QoS commands to ports in a LAG, the rules can be different according the configuration as described in the following:

- Port-level QoS configurations where QoS values are applied directly to the port. These commands include the following: **priority**, **priority-force**, **drop-precedence** and **drop-precedence force** .
- Port-level QoS configurations using commands that begin with the **qos** keyword. These commands include: **qos dscp decode-policy**, **qos pcp decode-policy**, **qos exp decode-policy**, **qos dscp force**, **qos pcp force**, **qos exp force**, **qos dscp encode on** and **qos pcp encode on**.

LAG configuration rules where QoS values are applied directly to the port

In port-level QoS Configurations where QoS Values are applied directly to the port, the considerations listed below must be followed.

1. Each port that is configured into the LAG must have the same **priority**, **priority-force**, **drop-precedence** and **drop-precedence force** configuration.

   If you try to configure a LAG with ports that have a different configuration for these commands, the LAG deployment will fail and you will get an error message as shown in the following.

   ```
   device(config)# lag mylag static
   device(config-lag-mylag)# ports eth 10/1 to 10/2
   device(config-lag-mylag)# primary 10/1
   device(config-lag-mylag)# deploy
   port 10/1 priority is 5, but port 10/2 priority is 0
   Error: port 10/1 and port 10/2 have different configurations
   LAG mylag deployment failed!
   device(config-lag-mylag)#
   ```
2. If you have already formed a LAG with the same configuration, you can change the configuration by making changes to the LAG's primary port.
3. If the LAG configuration is deleted, each of the port in the LAG (primary and secondary) will inherit the QoS configuration of the primary port.

LAG configuration rules for QoS configurations using commands that begin with the qos keyword

NOTE
Due to hardware considerations on the Brocade NetIron CER and Brocade NetIron CES devices, the encode policy must be applied on the ingress interface. This restriction results in the encode policies being applied to all packets coming in on the ingress port, regardless of egress port. See example configuration below.

In port-level QoS configurations where QoS configurations using commands that begin with the **qos** keyword are used, the considerations listed below must be followed.

1. The secondary ports configured in the LAG must not have any QoS values configured on them.
2. The **qos** commands that are configured on the primary port are applied to all ports in the LAG.
3. Once the LAG is formed, you can change the QoS configuration for all ports in the LAG by making changes to the LAG's primary port and you cannot make changes to the QoS configurations directly to any of the secondary ports.
4. If the LAG is deleted, the QoS configuration will only be retained on the primary port.

Example configuration

With the **qos dscp encode-policy** on and access-list 1302 applied, all packets coming in on interface e 2/1 and VLAN 11 will have a modified dscp value of 60 on the egress regardless of the egress port. In this example, packets not matching the ACL would have the port value set, per Forcing the drop precedence of a packet on page 90.

```
device(config)# interface ethernet 2/1
device(config)# qos dscp encode-policy on
device(config)# enable
device(config)# priority force
device(config)# priority 5
device(config)# drop-precedence force
device(config)# drop-precedence 3
device(config)# mac access-group 1302 in
!
access-list 1302 permit any any 11 etype any dscp-marking 60
!
```

Configuring port-level QoS commands on CPU ports

The control packets destined to the CPU are assigned fixed priorities. The data and control packets that are processed by the CPU are prioritized, scheduled, and rate-shaped so that the higher priority control packets are handled before any lower priority control and data packets. The enhanced control packet prioritization and scheduling scheme ensures the proper transmission or reception of time-sensitive control and protocol packets.

TABLE 16 Prioritized protocol and data packets

Priority categorization	Protocols
P7	LACP, UDLD, STP, RSTP, BPDU, VSRP, MRP, LLDP, VRRP, VRRP-E, 802.1x, FDP, CDP, BFD, ERP, 802.1ag.
P6	OSPF, IS-IS, RIP, RIPNG, BGP, IPv6 Neighbor Discovery, CCP, LDP, RSVP.
P5	PIM, PIM-DM, IGMP, MLD.
P4	ARP, DHCP, BOOTP.
P3	Telnet, SNMP.
P2	Reserved.
P1	sFlow.
P0	Data packets.

To configure a CPU port, enter the following command.

```
device(config)# cpu-port
device(config-cpu-port)#
```

Syntax: [no] cpu-port

The **no** option is used to disable QoS on the CPU port.

Configuring port-based rate shaping

You can limit the amount of bandwidth available on a CPU port by configuring the rate shaping value for that port. To set the rate shaping value, enter the following command.

```
device(config-cpu-port)# shaper 1000000
```

Syntax: [no] shaper *rate*

The *rate* parameter sets the rate shaping value for the CPU port. The acceptable value can be from 1 through 1000000 kilobits per second (Kbps).

The **no** option is used to reset the port shaping rate.

Configuring port and priority-based rate shaping

You can limit the amount of bandwidth available for a specified priority configured on a CPU port by configuring the rate shaping value for that priority. To set the rate shaping value for a priority queue, enter the following command.

```
device(config-cpu-port)# shaper priority 2 1000000
```

Syntax: [no] shaper priority *pri rate*

The *pri* parameter specifies the priority value for the configured port shaper. The priority can be from 0 through 7.

The *rate* parameter sets the rate shaping value for the priority queue configured on the CPU port. The acceptable value can be from 1 through 1000000 Kbps.

The **no** option is used to reset the port shaping rate for the priority queue.

Configuring traffic scheduling

Traffic scheduling can be configured on a per-port and per-queue basis. Traffic scheduling affects the outgoing traffic on the configured CPU port when bandwidth congestion occurs on that port. One strict profile, three Weighted Round Robin (WRR) profiles, and four mixed profiles define the scheduling attributes.

Configuring strict priority-based traffic scheduling

To configure strict priority-based scheduling, enter the following command.

```
device(config-cpu-port)# scheduler strict
```

Syntax: [no] scheduler strict

Strict priority-based scheduling is the default traffic scheduling scheme.

The **no** option is used to disable the strict scheduler.

Configuring WRR weight-based traffic scheduling

When configuring a CPU port scheduler to WRR, one of eight global scheduler profiles is taken and you can assign weights to each priority queue in global configuration mode.

To configure WRR weight-based scheduling, enter the following command.

```
device(config-cpu-port)# scheduler profile WRR0 weighted 40 10 20 30 40 50 60 10
```

Syntax: [no] scheduler profile *profilename* **weighted** [*queue7-weight queue6-weight queue5-weight queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*]

The *profile name* parameter specifies the name of the weighted scheduler.

The **weighted** keyword defines the weighted scheduler.

The *queue7-weight* parameter defines the value for queue 7 in calculating the allocated bandwidth of queue 7.

The *queue6-weight* parameter defines the value for queue 6 in calculating the allocated bandwidth of queue 6.

The *queue5-weight* parameter defines the value for queue 5 in calculating the allocated bandwidth of queue 5.

The *queue4-weight* parameter defines the value for queue 4 in calculating the allocated bandwidth of queue 4.

The *queue3-weight* parameter defines the value for queue 3 in calculating the allocated bandwidth of queue 3.

The *queue2-weight* parameter defines the value for queue 2 in calculating the allocated bandwidth of queue 2.

The *queue1-weight* parameter defines the value for queue 1 in calculating the allocated bandwidth of queue 1.

The *queue0-weight* parameter defines the value for queue 0 in calculating the allocated bandwidth of queue 0.

The **no** option is used to return to the default traffic scheduler.

Configuring mixed strict priority- and weight-based traffic scheduling

When configuring the mixed strict priority- and weight-based scheduling scheme, queue 5 to queue 7 are allocated to strict priority-based scheduling and queue 0 to queue 4 are allocated to weight-based scheduling.

To configure mixed strict priority- and weight-based scheduling., enter the following command.

```
device(config-cpu-port)# scheduler profile MIXED0 mixed 20 30 15 5 10
```

Syntax: [no] scheduler profile *profilename* **mixed** [*queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*]

The *profile name* parameter specifies the name of the mixed scheduler.

The **mixed** keyword defines the mixed scheduler.

The *queue4-weight* parameter defines the value for queue 4 in calculating the allocated bandwidth of queue 4.

The *queue3-weight* parameter defines the value for queue 3 in calculating the allocated bandwidth of queue 3.

The *queue2-weight* parameter defines the value for queue 2 in calculating the allocated bandwidth of queue 2.

The *queue1-weight* parameter defines the value for queue 1 in calculating the allocated bandwidth of queue 1.

The *queue0-weight* parameter defines the value for queue 0 in calculating the allocated bandwidth of queue 0.

The **no** option is used to return to the default traffic scheduler.

Displaying QoS information

You can display the following QoS information as described:

- **QoS Configuration Information** - Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can display the priority and drop-precedence values mapped between values internal to the device and values that are received at the device or marked on packets leaving the device.
- **QoS Packet and Byte Statistics** - Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can enable and display the contents of the QoS Packet and Byte Counters.

Displaying QoS Configuration information

You can display the following QoS Configuration information:

- QoS Decode Policy Map Configurations
- QoS Policy Map Binding Configurations

Clearing the QoS packet and byte counters

You can clear the QoS counters whose display is generated using the **clear np statistics** command.

Syntax: clear np statistics

Scheduling traffic for forwarding

If the traffic being processed by a Brocade device is within the capacity of the device, all traffic is forwarded as received. Once we reach the point where the device is bandwidth constrained, it becomes subject to traffic scheduling as described in this section.

The Brocade devices classify packets into one of eight internal priorities. Traffic scheduling allows you to selectively forward traffic according to the forwarding queue that is mapped to according to one of the following schemes:

- **Strict priority-based scheduling** - This scheme guarantees that higher-priority traffic is always serviced before lower priority traffic. The disadvantage of strict priority-based scheduling is that lower-priority traffic can be starved of any access.
- **WRR (Weighted Round Robin)** - With WRR destination-based scheduling enabled, some weight-based bandwidth is allocated to all queues. With this scheme, the configured weight distribution is guaranteed across all traffic leaving an egress port and an input port is guaranteed allocation in relationship to the configured weight distribution.
- **Mixed strict priority and weight-based scheduling** - This scheme provides a mixture of strict priority for the three highest priority queues and WRR for the remaining priority queues.

Configuring traffic scheduling

Traffic scheduling can be configured on a per-port basis. It affects the outgoing traffic on the configured port when bandwidth congestion occurs on that port. The following sections describe how to configure each of the traffic scheduling schemes.

Configuring strict priority-based traffic scheduling

To configure strict priority-based scheduling use a command such as the following.

```
device(config)# interface ethernet 1/1
device(config-if-e1000-1/1)# qos scheduler strict
```

Syntax: [no] qos scheduler strict

This is the default when traffic scheduling is not configured.

Calculating the values for WRR weight-based traffic scheduling

WRR (Weighted Round Robin) scheduling is configured to be a percentage of available bandwidth using the following formula. Remember weight is a relative value.

```
                         q (x)
Weight of q (x) = ------------------------------------
                   q0 + q1 + q2 +q3+ q4 + q5 +q6 +q7
```

Weight of q (x) = the calculated weight as a percentage of the port's total bandwidth.For example if you assign the following values to queues 0 to 7:

- Queue 0 =10, Queue 1 = 15, Queue 2 = 20, Queue 3 = 25, Queue 4 = 30, Queue 5 = 35, Queue 6 = 40, and Queue 7 = 45

Where:

q (x) = The value of the queue that you want to determine the weight for. It can be the value of any queue (0 - 7).

q0 - q7 = the assigned values of the eight queues.

Weight of q (x) = the calculated weight as a percentage of the port's total bandwidth.

Determining the correlation between weight and bandwidth example

To determine the correlation between weight and bandwidth use following calculation example.

If you assign the following values to queues 0 to 7:

- Queue 0 =10, Queue 1 = 15, Queue 2 = 20, Queue 3 = 25, Queue 4 = 30, Queue 5 = 35, Queue 6 = 40, and Queue 7 = 45,

To determine the weight of **q3**

```
                       25
Weight of q3 = -----------------------------------------
                10 + 15 + 20 + 25 + 30 + 35 + 40 + 45
```

The weight of q3 is 11.4%. Consequently, q3 will get 11.4% of the port's total bandwidth.

The values of the remaining queues are calculated to be the following: q7 = 20.5%, q6 = 18.2%, q5 = 15.9%, q4 = 13.6%, q3 = 11.4%, q2 = 9.1%, q1 = 6.8%, and q0 = 4.5%

NOTE
The easiest way to calculate the correlation between weight and bandwidth is to have the total weight of all queues sum up to 100, then the weight itself will be the % bandwidth available for that queue.

Configuring WRR weight-based traffic scheduling

To configure WRR weight-based scheduling use a command such as the following at the global configuration level.

```
device(config)# qos scheduler profile MIXED0 mixed 20 30 15 5 10
```

Syntax: qos scheduler profile *profile-name* **weighted** *queue7-weight queue6-weight queue5-weight queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*

The **weighted** parameter defines the weighted scheduler.

The **mixed** parameter defines the mixed scheduler.

The *queue7-weight* variable defines the relative value for queue7 in calculating queue7's allocated bandwidth.

The *queue6-weight* variable defines the relative value for queue6 in calculating queue6's allocated bandwidth.

The *queue5-weight* variable defines the relative value for queue5 in calculating queue5's allocated bandwidth.

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The*queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth.

Refer to Calculating the values for WRR weight-based traffic scheduling on page 98 for information on assigning queue0-weight to queue4-weight values.

Configuring mixed strict priority and weight-based scheduling

When configuring the mixed strict priority and weight-based scheduling option, queues 5 - 7 are allocated to strict priority-based scheduling and queues 0 - 4 are allocated to weight-based scheduling.

To configure mixed priority and weight-based scheduling use a command such as the following.

```
device(config)# qos scheduler profile Mixed0 mixed 100 80 60 40 20
```

Syntax: [no] qos scheduler profile *profile-name* **mixed** *queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The*queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth.

The acceptable range for *queuex-weight* variables is 1-128.

Refer to Calculating the values for WRR weight-based traffic scheduling on page 98 for information on assigning queue0-weight to queue4-weight values.

Egress port and priority based rate shaping

Rate shaping is a mechanism to smooth out the variations in traffic above a certain rate. The primary difference between rate shaping and rate limiting is that in rate limiting, traffic exceeding a certain threshold is dropped. In rate shaping, the traffic that exceeds a threshold is buffered so that the output from the buffer follows a more uniform pattern. Rate shaping is useful when burstiness in the source stream needs to be smoothed out and a more uniform traffic flow is expected at the destination.

NOTE
Because excess traffic is buffered, rate shaping must be used with caution. In general, it is not advisable to rate shape delay-sensitive traffic.

Brocade devices support egress rate shaping. Egress rate shaping is supported per port or for each priority queue on a specified port.

Configuring port-based rate shaping

When setting rate shaping for a port, you can limit the amount of bandwidth available on a port within the limits of the port's rated capacity.

NOTE
The egress rate shaping on a port-based and priority based rate shaper is configured in increments of 1Kbps

These limits provide a minimum and maximum rate that the port can be set to. They also provide the increments at which the port capacity can be set. In operation, you can set any number between the minimum and maximum values. The device will automatically round-up the value to the next higher increment.

To set a 10 Gbps port to the incremental port capacity over 2 Gbps, use the following command.

```
device(config)# interface ethernet 2/2
device(config-if-e10000-2/2)# qos shaper 2000000
```

Syntax: [no] qos shaper *rate*

The *rate* variable sets the rate you want to set for the port within the limits available.

Configuring port and priority-based rate shaping

When setting rate shaping for a priority queue, you can limit the amount of bandwidth available for a specified priority within the limits of the capacity of the port that the priority is configured on. You can set the limit for the priority to any value from one to the port's maximum rating and the device will automatically round-up the value to the next increment supported. This will be a slightly higher value than what you specify with the command. For example, if you set the rate for priority 2 on a 10G port to 2,000,000,100, the actual rate would be slightly higher.

NOTE
The egress rate shaping burst size for a port and priority-based shaper is 4096 bytes.

To set the capacity for priority 2 traffic on a 10 Gbps port to the incremental capacity over 2 Gbps, use the following command.

```
device(config)# interface ethernet 2/2
device(config-if-e10000-2/2)# qos shaper priority 2 2000000
```

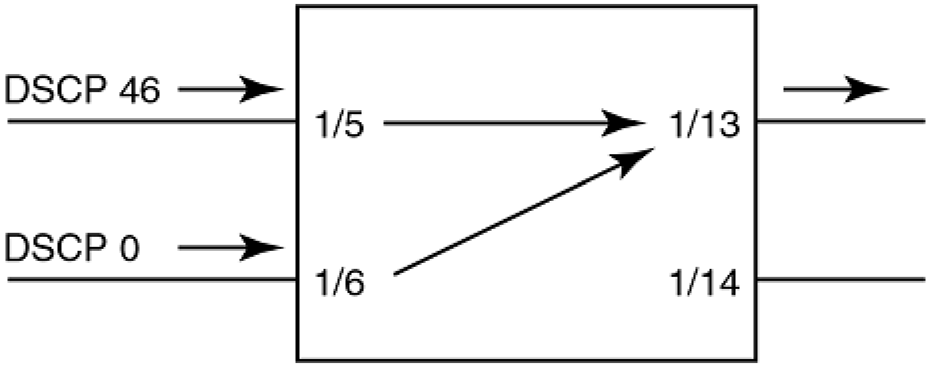
Syntax: [no] qos shaper priority *priority-level rate*

The *priority-level* variable specifies the priority that you want to set rate shaping for on the port being configured.

The *rate* variable sets the rate you want to set for the priority.

Example of configuring Prioritized Voice over Data

When configuring Prioritized Voice over Data, use the strict priority method.



In the example below, the DSCP 46 (Voice) is assigned to high priority queue on Ingress port, and goes out of Egress port without changing the DSCP value in the packet.

```
device(config)#int e 1/5
device(config-if-e1000-1/5)#qos scheduler strict
device(config-if-e1000-1/5)#int e 1/6
device(config-if-e1000-1/6)#qos scheduler strict
device(config-if-e1000-1/6)#int e 1/13
device(config-if-e1000-1/13)#qos dscp encode-policy off
device(config-if-e1000-1/13)#qos scheduler strict
```

Use the **show qos-map dscp decode-map default-map** command to verify DSCP 46 is going to the high priority queue as highlighted below.

```
device(config)#show qos-map dscp decode-map default-map
DSCP decode default-map
  DSCP  0 to priority  0 drop-precedence  0
  DSCP  1 to priority  0 drop-precedence  0
  DSCP  2 to priority  0 drop-precedence  1
  DSCP  3 to priority  0 drop-precedence  1
  DSCP  4 to priority  0 drop-precedence  2
  DSCP  5 to priority  0 drop-precedence  2
  DSCP  6 to priority  0 drop-precedence  3
  DSCP  7 to priority  0 drop-precedence  3
  DSCP  8 to priority  1 drop-precedence  0
  DSCP  9 to priority  1 drop-precedence  0
  DSCP 10 to priority  1 drop-precedence  1
  DSCP 11 to priority  1 drop-precedence  1
  DSCP 12 to priority  1 drop-precedence  2
  DSCP 13 to priority  1 drop-precedence  2
  DSCP 14 to priority  1 drop-precedence  3
  DSCP 15 to priority  1 drop-precedence  3
  DSCP 16 to priority  2 drop-precedence  0
  DSCP 17 to priority  2 drop-precedence  0
  DSCP 18 to priority  2 drop-precedence  1
  DSCP 19 to priority  2 drop-precedence  1
  DSCP 20 to priority  2 drop-precedence  2
  DSCP 21 to priority  2 drop-precedence  2
  DSCP 22 to priority  2 drop-precedence  3
  DSCP 23 to priority  2 drop-precedence  3
  DSCP 24 to priority  3 drop-precedence  0
  DSCP 25 to priority  3 drop-precedence  0
  DSCP 26 to priority  3 drop-precedence  1
  DSCP 27 to priority  3 drop-precedence  1
  DSCP 28 to priority  3 drop-precedence  2
  DSCP 29 to priority  3 drop-precedence  2
  DSCP 30 to priority  3 drop-precedence  3
  DSCP 31 to priority  3 drop-precedence  3
  DSCP 32 to priority  4 drop-precedence  0
  DSCP 33 to priority  4 drop-precedence  0
  DSCP 34 to priority  4 drop-precedence  1
  DSCP 35 to priority  4 drop-precedence  1
  DSCP 36 to priority  4 drop-precedence  2
  DSCP 37 to priority  4 drop-precedence  2
  DSCP 38 to priority  4 drop-precedence  3
  DSCP 39 to priority  4 drop-precedence  3
  DSCP 40 to priority  5 drop-precedence  0
  DSCP 41 to priority  5 drop-precedence  0
  DSCP 42 to priority  5 drop-precedence  1
  DSCP 43 to priority  5 drop-precedence  1
  DSCP 44 to priority  5 drop-precedence  2
  DSCP 45 to priority  5 drop-precedence  2
  DSCP 46 to priority  5 drop-precedence  3
  DSCP 47 to priority  5 drop-precedence  3
  DSCP 48 to priority  6 drop-precedence  0
  DSCP 49 to priority  6 drop-precedence  0
  DSCP 50 to priority  6 drop-precedence  1
  DSCP 51 to priority  6 drop-precedence  1
  DSCP 52 to priority  6 drop-precedence  2
  DSCP 53 to priority  6 drop-precedence  2
  DSCP 54 to priority  6 drop-precedence  3
  DSCP 55 to priority  6 drop-precedence  3
  DSCP 56 to priority  7 drop-precedence  0
  DSCP 57 to priority  7 drop-precedence  0
  DSCP 58 to priority  7 drop-precedence  1
  DSCP 59 to priority  7 drop-precedence  1
  DSCP 60 to priority  7 drop-precedence  2
  DSCP 61 to priority  7 drop-precedence  2
  DSCP 62 to priority  7 drop-precedence  3
  DSCP 63 to priority  7 drop-precedence  3
device(config)#
```

Syntax: show qos-map dscp decode-map default-map

If you want to change from strict to any other policy for weighted or mixed on the Brocade NetIron CES and Brocade NetIron CER, enter the **show qos scheduler profile** command as show below.

```
device#show qos scheduler profile
Index  | Scheme    Type   Pri7   Pri6   Pri5   Pri4   Pri3   Pri2   Pri1   Pri0
-------+----------------+------+------+------+------+------+------+------+----
     0 | Strict
     1 | WRR0      Weight 40     10     20     30     40     50     60     10
     2 | WRR1      Weight 10     10     10     10     10     10     10     10
     3 | WRR2      Weight 10     10     10     10     10     10     10     10
     4 | MIXED0    Weight                      100    80     60     40     20
     5 | MIXED1    Weight                      20     40     60     80     100
     6 | MIXED2    Weight                      30     30     30     30     30
     7 | MIXED3    Weight                      128    128    1      1      1
```

Syntax: show qos scheduler profile

To change the profile from strict to weighted WRR0, use the **qos scheduler WRR0** command on the interface (in this case you would change it on the Outgoing interface ie 1/13. Remember, on the Brocade NetIron CES and Brocade NetIron CER the QOS Scheduler is on Egress interface only.

```
device(config)#sh run int e 1/13
interface ethernet 1/13
 qos scheduler profile WRR0
 enable
```

To change the settings of weighted WRR0, use the **qos scheduler profile WRR0** command in global config mode.

```
device(config)# qos scheduler profile WRRO weighted 40 10 20 30 40 50 60 10
```

Syntax: qos scheduler profile

Issue the **show qos scheduler profile** command to verify the change.

```
device#show qos scheduler profile
Index  | Scheme    Type   Pri7   Pri6   Pri5   Pri4   Pri3   Pri2   Pri1   Pri0
-------+---------------+------+------+------+------+------+------+------+----
     0 | Strict
     1 | WRR0      Weight 40     10     20     30     40     50     60     10
     2 | WRR1      Weight 10     10     10     10     10     10     10     10
     3 | WRR2      Weight 10     10     10     10     10     10     10     10
     4 | MIXED0    Weight                      100    80     60     40     20
     5 | MIXED1    Weight                      20     40     60     80     100
     6 | MIXED2    Weight                      30     30     30     30     30
     7 | MIXED3    Weight                      128    128    1      1      1
```

Syntax: show qos scheduler profile

Clearing traffic manager statistics

You can clear statistic for a Brocade device as shown in the following.

```
device# clear statistics
```

Syntax: clear statistics

New network processor counters displayed

Output from the **show interface** command has been enhanced to provide the following related information:

- The number of packets received at the network processor (NP)
- The number of packets sent from the NP
- The Number of ingress packets dropped at the NP
- The number of packets transmitted from the NP
- The number of packets received by the NP

The following is an example of the new output from the **show interface** command.

```
device(config)# show interface ethernet 3/3
GigabitEthernet3/3 is up, line protocol is up
Hardware is GigabitEthernet, address is 0004.80a0.4052 (bia 0004.80a0.4052)
Configured speed auto, actual 1Gbit, configured duplex fdx, actual fdx
Member of L2 VLAN ID 1, port is untagged, port state is Forwarding
STP configured to ON, Priority is level0, flow control enabled
mirror disabled, monitor disabled
Not member of any active trunks
Not member of any configured trunks
No port name
MTU 1544 bytes, encapsulation ethernet
300 second input rate: 754303848 bits/sec, 1473249 packets/sec, 89.57% utilization
300 second output rate: 754304283 bits/sec, 1473250 packets/sec, 89.57% utilization
1015230949 packets input, 64974783168 bytes, 0 no buffer
Received 0 broadcasts, 0 multicasts, 1015230949 unicasts
0 input errors, 0 CRC, 0 frame, 0 ignored
0 runts, 0 giants
NP Ingress dropped 0 packets
1015231660 packets output, 64974824768 bytes, 0 underruns
Transmitted 0 broadcasts, 0 multicasts, 1015231660 unicasts
0 output errors, 0 collisions
```

Configuring Quality of Service for the Brocade NetIron XMR and Brocade NetIron MLX Series

This chapter describes how QoS is implemented and configured in the Brocade device.

NOTE
Internet Protocol Television (IPTV) multicast streams are supported on Brocade NetIron CES and Brocade NetIron CER devices.

Ingress Traffic processing through a device

The QoS operation on Ingress Traffic of a Brocade device involves reception and processing of packets based upon priority information contained within the packet. As the packets are processed through the device, there are several opportunities to influence the processing by configuration as described in the steps below.

1. Derive priority and drop precedence from the packets PCP (IEEE 802.1p) value. The Priority Code Point (PCP) is a 3-bit field within an IEEE 802.1Q tagged frame that is used to convey the priority of the frame. By using a mapping table, the 3-bit PCP field can be decoded to derive priority and drop precedence information.

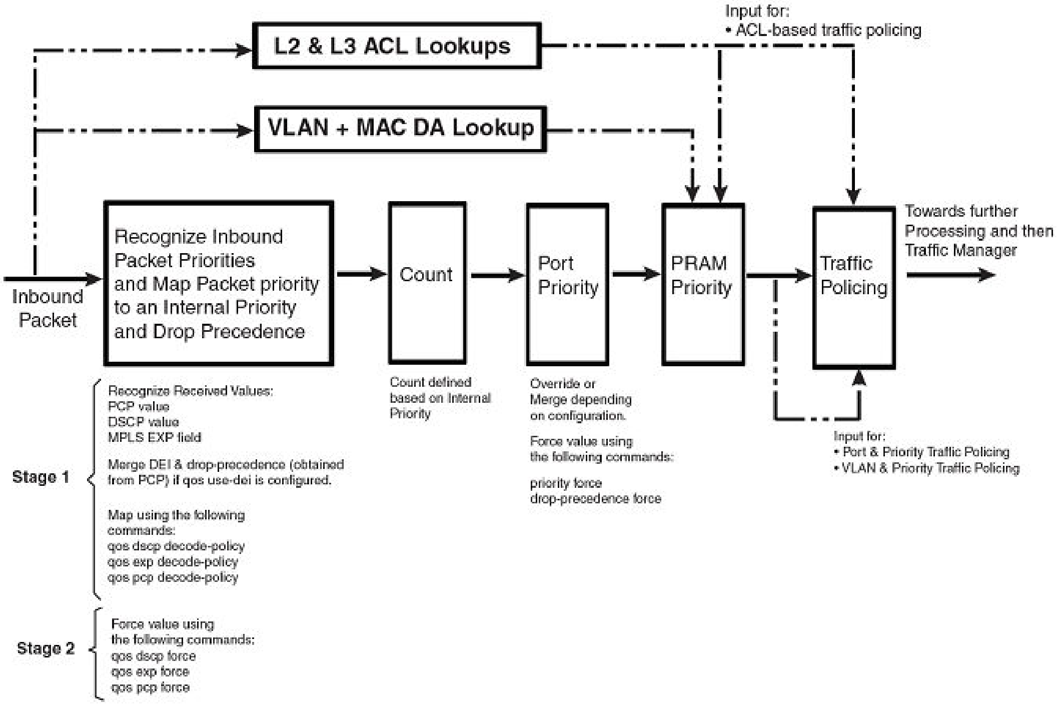
NOTE
The PCP field was formerly called IEEE 802.1p.

2. Derive priority and drop precedence from the packets EXP value.
3. Derive priority and drop precedence from the packets DSCP value.

NOTE
DSCP encoding and decoding are not supported for MPLS packets.

4. Merge or force the priorities described in steps 1 through 3.
5. Merge or force the priority and drop precedence value based on the value configured for the physical port.
6. Merge or force the priority value based on the value configured for the VLAN.
7. Merge or force the priority value based on an ACL look-up. This is used for setting a a specific priority for and L2, L3 or L4 traffic flow.

FIGURE 1 Logic flow of Ingress QoS processing


Input for:
• ACL-based traffic policing
L2 & L3 ACL Lookups
VLAN + MAC DA Lookup
Inbound Packet
Recognize Inbound Packet Priorities and Map Packet priority to an Internal Priority and Drop Precedence
Count
Port Priority
PRAM Priority
Traffic Policing
Towards further Processing and then Traffic Manager
Recognize Received Values:
PCP value
DSCP value
MPLS EXP field
Merge DEI & drop-precedence (obtained from PCP) if qos use-dei is configured.
Map using the following commands:
qos dscp decode-policy
qos exp decode-policy
qos pcp decode-policy
Stage 1
Stage 2
Force value using the following commands:
qos dscp force
qos exp force
qos pcp force
Count defined based on Internal Priority
Override or Merge depending on configuration.
Force value using the following commands:
priority force
drop-precedence force
Input for:
• Port & Priority Traffic Policing
• VLAN & Priority Traffic Policing

Recognizing inbound packet priorities and mapping to internal priority

The logic flow of Ingress QoS processing can be described in two stages:

Stage 1 Collect priority and drop precedence information from various portions of the packet header

- If a packet's EtherType matches 8100 or the port's EtherType, derive a priority value and drop precedence by decoding the PCP value.
- If the **qos use-dei** command is configured, the bit between the VLAN ID and PCP in the VLAN tag will be interpreted as a drop precedence and priority value.
- For MPLS packets, derive a priority value and drop precedence by decoding the EXP bits.
- For IPv4 or v6 packets, derive a priority value and drop precedence by decoding the DSCP bits.
- The derived values for PCP, EXP and DSCP are mapped to either a default map or a configured Ingress Decode Policy Map.
- To assist the device in the decoding process described in "stage 1" decode-map tables are defined.

Stage 2 Determine if a priority value should be forced or merged

- If a packet's EtherType matches 8100 or the port's EtherType, derive a priority value and drop precedence by decoding the PCP value.
- If the **qos pcp force** command is configured on the port, the priority and drop precedence values are set to the value read from the PCP bits.
- If the **qos exp force** command is configured on the port, the priority and drop precedence values are set to the value read from the MPLS EXP bits.
- If the **qos dscp force** command is configured on the port, the priority and drop precedence values are set to the value read from the DSCP bits.
- If none of the qos force commands are configured, the priority and drop precedence values are set for IPv4 or v6 packets and MPLS packets as described in the following:

  For IPv4 or v6 Packets: Priority and drop precedence values obtained as a merge of the decoded PCP and decoded DSCP values.

  For MPLS Packets: Priority and drop precedence values obtained as a merge of the decoded PCP and decoded EXP values.

Creating an Ingress decode policy map

Once a packet's Ingress priority has been recognized for the PCP, DSCP and EXP values, those values are matched against a policy map to determine the priority and drop precedence values that will be assigned to the packet within the device. The maps used can be either:

- Default policy maps described in Default QoS mappings on page 112.
- User-configured policy maps that are defined as described:

  dscp decode-map *decode-map-name* - This command allows you to map a recognized DSCP value to a value that you define.

  pcp decode-map *decode-map-name* - This command allows you to map a recognized PCP value to a value that you define.

  exp decode-map *decode-map-name* - This command allows you to map a recognized MPLS EXP value to a value that you define.

Forcing or merging the priority of a packet

Once a packet’s Ingress priority has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to "force" the priority of a packet based on the following criteria:

- Forced to a priority configured for a specific Ingress port. The **priority force** command is configured at the interface where you want it to be applied.
- Forced to a priority configured for a specific VLAN. The **priority force** command is configured at the VLAN where you want is to be applied.
- Forced to a priority that is obtained from the DSCP priority bits. The **qos- dscp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the EXP priority bits. The **qos- exp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the PCP priority bits. The **qos- pcp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is based on an ACL match. The **priority-force** keyword can be used within an ACL to apply a priority to specified traffic.

If multiple commands containing the **priority-force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

1. ACL match (if the **qos-tos mark cos** command is configured, it has the same #1 priority precedence as ACL match). Refer to Specifying the trust level and enabling marking on page 143 for details.
2. VLAN priority

NOTE
VLAN priority works differently for Layer 2 and Layer 3 traffic. If you apply VLAN priority to a physical port for layer 2 traffic then it will apply VLAN priority and will change traffic accordingly. However, if you apply VLAN priority to a physical port for layer 3 traffic then it will not apply and will not change traffic. VLAN priority will only apply to layer 3 traffic if the port is a VE port.

3. Physical port priority value.
4. DSCP value in an incoming IPv4 or IPv6 packet.
5. EXP value in an incoming MPLS packet.
6. PCP value in a tagged frame or PCP field or .1ad DE

Details of how to configure the force commands are provided in Configuring a force priority on page 132.

Forcing or merging the drop precedence of a packet

Once a packet’s Ingress drop precedence has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to "force" the drop precedence of a packet based on the following criteria:

- Forced to a drop precedence configured for a specific Ingress port. The **drop-precedence force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the DSCP priority bits. The **qos dscp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the EXP priority bits. The **qos exp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the PCP priority bits. The **qos pcp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is based on an ACL match. The **drop-precedence-force** keyword can be used within an ACL to apply a priority to specified traffic.

If multiple commands containing the **force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

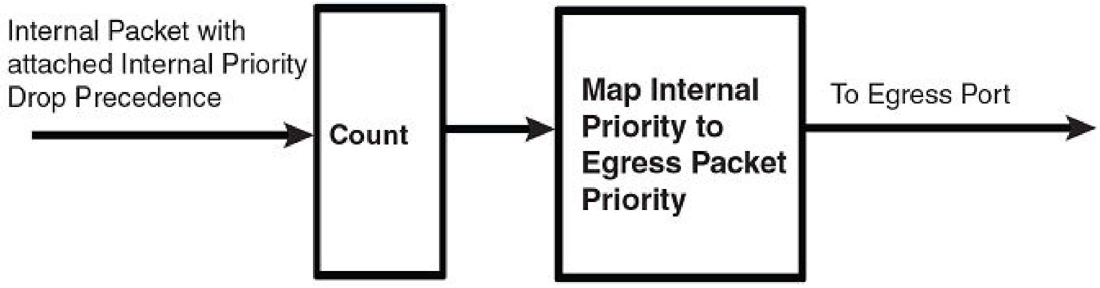
1. ACL match
2. Physical port's drop precedence value
3. DSCP value in an incoming IPv4 or IPv6 packet
4. EXP value in an incoming MPLS packet
5. PCP value in a tagged frame or PCP field or .1ad DE

Details of how to configure the force commands are provided in Configuring a force priority on page 132.

Egress Traffic processing exiting a device

The QoS operation on Egress Traffic of a Brocade device involves marking packets as they leave a device on the egress port. As the packets are prepared to exit the device you can set the PCP, DSCP, and EXP values in the packet headers.

FIGURE 2 Logic flow of Egress QoS processing



Creating an egress encode policy map

The QoS value that a packet carries in its header when it exits a Brocade device on an egress interface is determined by a specified mapping. Unless configured, this value once determined is placed in an internal queue by using one of the default maps described in Default QoS mappings on page 112. Alternately, the following commands can be used to define an alternate mapping:

- **pcp encode-map** *encode-map-name* - This command allows you to map the internal priority and drop precedence values of a packet into the PCP code point.
- **dscp encode-map** *encode-map-name* - This command allows you to map the internal priority and drop precedence values of a packet into the DSCP code point.

NOTE
DSCP encoding and decoding are not supported for MPLS packets.

- **exp encode-map** *encode-map-name* - This command allows you to map the internal priority and drop precedence values of a packet into the EXP code point.

Backward compatibility with pre-03.8.00

A number of the commands used in prior releases for QoS configuration have been deprecated and the functions performed by them has been taken over by new commands. The **qos-tos-trust** and **qos-tos mark** commands are still operative although their use is discouraged. Additionally, the **qos-tos map dscp-priority** commands that are in a current configuration are converted to new commands during a software upgrade.

Commands deprecated in version 03.8.00

The following table describes each of the commands that were deprecated with version 3.8.00, the purpose of the command, and the equivalent functionality available in versions 03.8.00 and later.

TABLE 17 Deprecated commands from pre-03.8.00 releases

Pre-03.8.00command	Purpose of the pre-03.8.00 command	Equivalent in advanced QoS infrastructure
`port-priority`	Initializes the global TOS or DSCP table. Once executed, the priority from the DSCP bits is merged with other priorities **NOTE** By default, DSCP processing is ignored.	• Default behavior is now to process DSCP. • The global DSCP table is now extended into a per-port DSCP table. • If not desired use the following command: `qos dscp encode-policy all-zero-map`
`qos-tos` at the interface level	Initializes the global TOS or DSCP table. Once executed, the priority from the DSCP bits is merged with other priorities	• Default behavior is now to process DSCP. • The global DSCP table is now extended into a per-port DSCP table. • If not desired use the following command at the physical interface level: `qos dscp encode-policy all-zero-map`

TABLE 17 Deprecated commands from pre-03.8.00 releases (Continued)

Pre-03.8.00command	Purpose of the pre-03.8.00 command	Equivalent in advanced QoS infrastructure
`merge-egress-priorities`	Allows a packet to be sent out with the higher of two possible values: • the initial IEEE 802.1p value that the packet arrived with • a new (higher) priority that the packet has been forced to.	This feature is no longer supported because it is not useful. Typically, if a packet is remarked, we want the remarked priority value to be reflected in the outgoing packet and not be a merge of two priorities.
`qos-tos map dscp-priority` at the global configuration level	Initializes entries in the global TOS or DSCP table.	These commands are deprecated and converted into a special decode map named: USER_DSCP_MAP. This process is described in detail in DSCP-priority mapping commands on page 111.

qos-tos trust and qos-tos mark commands

The **qos-tos trust** and **qos-tos mark** commands are retained in 03.8.00 and later versions of the Multi-Service IronWaresoftware as described in the following:

- The primary use of these commands is for packet remarking (without changing the internal priority of the packet if desired)

  qos-tos trust indicates the priority to be trusted on the Ingress interface (cos, dscp or ip-prec).

  qos-tos mark indicates which priority is to be marked on the Egress interface (cos or dscp).

- **qos-tos trust** and **qos-tos mark** commands are both applied at the Ingress interface (physical or virtual).

  For instructions concerning the configuration of these commands, refer to Specifying the trust level and enabling marking on page 143.

NOTE
Because these commands use L4 CAM entries, this may not be used in conjunction with L2 or L3 Multicast Forwarding, or other ACL's.

DSCP-priority mapping commands

In releases prior to 03.8.00, there is a command that globally maps the DSCP priorities in the incoming packets to an internal priority. This command: **qos-tos map dscp-priority** is not available in release 03.8.00 and later. If you have configured the **qos-tos map dscp-priority** command in a previous release, the operation of your device will not be changed by upgrading but the configuration commands in the configuration file will be automatically updated to the commands introduced in release 03.8.00. This will be done by the method described in the following:

- the mapping defined by the **qos-tos map dscp-priority** commands is converted to a qos-mapping configuration of a decode map titled USER_DSCP_MAP.
- If the **port-priority** command is configured, the **qos dscp encode-policy** command is included in the configuration and the decode map titled: USER_DSCP_MAP is applied globally on the device.

Mapping from qos-tos mapping to USER_DSCP_MAP

As described previously, in release 03.8.00 and later, a configuration containing **qos-tos map dscp-priority** commands is converted to a **qos-mapping** configuration of a decode map titled USER_DSCP_MAP. The following **qos-tos map dscp-priority** commands map various DSCP priority values to internal priorities from 0 to 7.

```
device(config)# port-priority
device(config)# qos-tos map dscp-priority 0 2 3 4 to 1
device(config)# qos-tos map dscp-priority 8 to 5
device(config)# qos-tos map dscp-priority 16 to 4
device(config)# qos-tos map dscp-priority 24 to 2
device(config)# qos-tos map dscp-priority 32 to 0
device(config)# qos-tos map dscp-priority 40 to 7
device(config)# qos-tos map dscp-priority 48 to 3
device(config)# qos-tos map dscp-priority 56 to 6
```

The following example displays the USER_DSCP_MAP configuration and binding.

```
qos-mapping
  dscp decode-map USER_DSCP_MAP
    dscp-value 32 to priority 0
    dscp-value 0 to priority 1
    dscp-value 2 3 to priority 1
    dscp-value 4 to priority 1
    dscp-value 24 to priority 2
    dscp-value 48 to priority 3
    dscp-value 16 to priority 4
    dscp-value 8 to priority 5
    dscp-value 56 to priority 6
    dscp-value 40 to priority 7
qos dscp decode-policy USER_DSCP_MAP
```

NOTE
If the **port-priority** command was not configured in the pre-converted configuration, the **qos dscp encode-policy USER_DSCP_MAP** command will not be added to the converted file and you will have to configure the **qos dscp encode-policy** command to bind "*USER_DSCP_MAP* " either globally or to a specified port.

Default QoS mappings

If a user defined map is not created or applied to Ingress or Egress traffic, the Brocade device uses a default map to assign PCP, DSCP and EXP priority and drop precedence values. The following tables describe the default QoS mapping values:

- PCP Encode Table
- PCP Decode Table
- DSCP Encode Table
- DSCP Decode Table
- EXP Encode Table
- EXP Decode Table

NOTE
The encode commands for QOS mapping such as **qos pcp encode** on | off, **qos dscp encode** on| off and **qos exp encode** on | off are available only at the physical port level. There are no direct commands at global level to enable or disable QOS encode mapping.

TABLE 18 Default PCP encode table

Priority &Drop Eligibility (DE)		7	7DE	6	6DE	5	5DE	4	4DE	3	3DE	2	2DE	1	1DE	0	0DE
PCP	8P0D(default)	7	7	6	6	5	5	4	4	3	3	2	2	1	1	0	0
	7P1D	7	7	6	6	5	4	5	4	3	3	2	2	1	1	0	0
	6P2D	7	7	6	6	5	4	5	4	3	2	3	2	1	1	0	0
	5P3D	7	7	6	6	5	4	5	4	3	2	3	2	1	0	1	0

TABLE 19 Default PCP decode table

PCP		7	6	5	4	3	2	1	0
PCP	8P0D(default)	7	6	5	4	3	2	1	0
	7P1D	7	6	4	4DE	3	2	1	0
	6P2D	7	6	4	4DE	2	2DE	1	0
	5P3D	7	6	4	4DE	2	2DE	0	0DE

TABLE 20 Default DSCP encode table

Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)	Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)
0 (000)	0 (00)	0 (000000)	4 (100)	0 (00)	32 (100000)
0 (000)	1 (01)	2 (000010)	4 (100)	1 (01)	34 (100010)
0 (000)	2 (10)	4 (000100)	4 (100)	2 (10)	36 (100100)
0 (000)	3 (11)	6 (000110)	4 (100)	3 (11)	38 (100110)
1 (001)	0 (00)	8 (001000)	5 (101)	0 (00)	40 (101000)
1 (001)	1 (01)	10 (001010)	5 (101)	1 (01)	42 (101010)
1 (001)	2 (10)	12 (001100)	5 (101)	2 (10)	44 (101100)
1 (001)	3 (11)	14 (001110)	5 (101)	3 (11)	46 (101110)
2 (010)	0 (00)	16 (010000)	6 (110)	0 (00)	48 (110000)
2 (010)	1 (01)	18 (010010)	6 (110)	1 (01)	50 (110010)
2 (010)	2 (10)	20 (010100)	6 (110)	2 (10)	52 (110100)
2 (010)	3 (11)	22 (010110)	6 (110)	3 (11)	54 (110110)

TABLE 20 Default DSCP encode table (Continued)

Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)	Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)
3 (011)	0 (00)	24 (011000)	7 (111)	0 (00)	56 (111000)
3 (011)	1 (01)	26 (011010)	7 (111)	1 (01)	58 (111010)
3 (011)	2 (10)	28 (011100)	7 (111)	2 (10)	60 (111100)
3 (011)	3 (11)	30 (011110)	7 (111)	3 (11)	62 (111110)

TABLE 21 Default DSCP decode table

DSCPdecimal (binary)	Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)	Prioritydecimal (binary)	Drop-precedence decimal (binary)
0 (000000)	0 (000)	0 (00)	16 (010000)	2 (010)	0 (00)
1 (000001)	0 (000)	0 (00)	17(010001)	2 (010)	0 (00)
2 (000010)	0 (000)	1 (01)	18 (010010)	2 (010)	1 (01)
3 (000011)	0 (000)	1 (01)	19 (010011)	2 (010)	1 (01)
4 (000100)	0 (000)	2 (10)	20 (010100)	2 (010)	2 (10)
5 (000101)	0 (000)	2 (10)	21(010101	2 (010)	2 (10)
6 (000110)	0 (000)	3 (11)	22 (010110)	2 (010)	3 (11)
7 (000111)	0 (000)	3 (11)	23 (010111)	2 (010)	3 (11)
8 (001000)	1 (001)	0 (00)	24 (011000)	3 (011)	0 (00)
9 (001001)	1 (001)	0 (00)	25 (011001)	3 (011)	0 (00)
10 (001010)	1 (001)	1 (01)	26 (011010)	3 (011)	1 (01)
11 (001011)	1 (001)	1 (01)	27 (011011)	3 (011)	1 (01)
12 (001100)	1 (001)	2 (10)	28 (011100)	3 (011)	2 (10)
13 (001101	1 (001)	2 (10)	29 (011101)	3 (011)	2 (10)
14 (001110)	1 (001)	3 (11)	30 (011110)	3 (011)	3 (11)
15 (001111)	1 (001)	3 (11)	31 (011111)	3 (011)	3 (11)

TABLE 22 Default DSCP decode table (cont.)

DSCPdecimal (binary)	Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)	Prioritydecimal (binary)	Drop-precedence decimal (binary
32 (100000)	4 (100)	0 (00)	48 (110000)	6 (110)	0 (00)
33 (100001)	4 (100)	0 (00)	49 (110001)	6 (110)	0 (00)
34 (100010)	4 (100)	1 (01)	50 (110010)	6 (110)	1 (01)
35 (100011)	4 (100)	1 (01)	51(110011)	6 (110)	1 (01)
36 (100100)	4 (100)	2 (10)	52(110100)	6 (110)	2 (10)
37 (100101)	4 (100)	2 (10)	53(110101)	6 (110)	2 (10)
38 (100110)	4 (100)	3 (11)	54 (110110)	6 (110)	3 (11)
38 (100111)	4 (100)	3 (11)	55 (110111)	6 (110)	3 (11)
40 (101000)	5 (101)	0 (00)	56 (111000)	7 (111)	0 (00)
41 (101001)	5 (101)	0 (00)	57 (111001)	7 (111)	0 (00)
42 (101010)	5 (101)	1 (01)	58 (111010)	7 (111)	1 (01)
43 (101011)	5 (101)	1 (01)	58 (111011)	7 (111)	1 (01)
44 (101100)	5 (101)	2 (10)	60 (111100)	7 (111)	2 (10)
45 (101101)	5 (101)	2 (10)	61 (111101)	7 (111)	2 (10)
46 (101110)	5 (101)	3 (11)	62 (111110)	7 (111)	3 (11)
47(101111)	5 (101)	3 (11)	63 (111111)	7 (111)	3 (11)

The following table lists the default EXP Encode mappings. Please note that software forwarded VPLS packets do not use the EXP encode table.

TABLE 23 Default EXP encode table

Prioritydecimal (binary)	Drop-precedence decimal (binary)	EXP value	Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)
0 (000)	0 (00)	0	4 (100)	0 (00)	4
0 (000)	1 (01)	0	4 (100)	1 (01)	4
0 (000)	2 (10)	0	4 (100)	2 (10)	4

TABLE 23 Default EXP encode table (Continued)

Prioritydecimal (binary)	Drop-precedence decimal (binary)	EXP value	Prioritydecimal (binary)	Drop-precedence decimal (binary)	DSCPdecimal (binary)
0 (000)	3 (11)	0	4 (100)	3 (11)	4
1 (001)	0 (00)	1	5 (101)	0 (00)	5
1 (001)	1 (01)	1	5 (101)	1 (01)	5
1 (001)	2 (10)	1	5 (101)	2 (10)	5
1 (001)	3 (11)	1	5 (101)	3 (11)	5
2 (010)	0 (00)	2	6 (110)	0 (00)	6
2 (010)	1 (01)	2	6 (110)	1 (01)	6
2 (010)	2 (10)	2	6 (110)	2 (10)	6
2 (010)	3 (11)	2	6 (110)	3 (11)	6
3 (011)	0 (00)	3	7 (111)	0 (00)	7
3 (011)	1 (01)	3	7 (111)	1 (01)	7
3 (011)	2 (10)	3	7 (111)	2 (10)	7
3 (011)	3 (11)	3	7 (111)	3 (11)	7

TABLE 24 Default EXP encode table

Prioritydecimal (binary)	Drop-precedence decimal (binary)	EXP value	Prioritydecimal (binary)	Drop-precedence decimal (binary)	EXP value
0 (000)	0 (00)	0	4 (100)	0 (00)	4
0 (000)	1 (01)	0	4 (100)	1 (01)	4
0 (000)	2 (10)	0	4 (100)	2 (10)	4
0 (000)	3 (11)	0	4 (100)	3 (11)	4
1 (001)	0 (00)	1	5 (101)	0 (00)	5
1 (001)	1 (01)	1	5 (101)	1 (01)	5
1 (001)	2 (10)	1	5 (101)	2 (10)	5
1 (001)	3 (11)	1	5 (101)	3 (11)	5
2 (010)	0 (00)	2	6 (110)	0 (00)	6

TABLE 24 Default EXP encode table (Continued)

Prioritydecimal (binary)	Drop-precedence decimal (binary)	EXP value	Prioritydecimal (binary)	Drop-precedence decimal (binary)	EXP value
2 (010)	1 (01)	2	6 (110)	1 (01)	6
2 (010)	2 (10)	2	6 (110)	2 (10)	6
2 (010)	3 (11)	2	6 (110)	3 (11)	6
3 (011)	0 (00)	3	7 (111)	0 (00)	7
3 (011)	1 (01)	3	7 (111)	1 (01)	7
3 (011)	2 (10)	3	7 (111)	2 (10)	7
3 (011)	3 (11)	3	7 (111)	3 (11)	7

TABLE 25 Default EXP decode table

EXP value	Prioritydecimal (binary)	Drop-precedence decimal (binary)
7	7 (111)	0
6	6 (110)	0
5	5 (101)	0
4	4 (100)	0
3	3 (011)	0
2	2 (010)	0
1	2 (001)	0
0	0 (000)	0

Protocol Packet Prioritization

Certain control packets are handled with certain priorities by default and hence those priorities cannot be lowered with any of the QoS configuration commands or the **priority force** command. The list of these control packets are listed below.

The following table lists the protocol packets that are internally and automatically prioritized for IPv4, Layer 2, and IPv6.

TABLE 26 Default prioritized protocol table

Protocol Packets
IPv4/L2
ARP
STP/RSTP/BPDU
MRP
VSRP
LACP
GARP
UDLD
IGMP
OSPF / OSPF over GRE
BGP / BGP over GRE
RIP
IS-IS
ES-IS
VRRP
VRRPE
PIM / PIM over GRE
MSDP / MSDP over GRE
RSVP
LDP basic
LDP extended
BOOTP/DHCP
IPv4 Router Alert
ISIS over GRE or GRE
Keep Alive Packets
BFD (Bidirectional Forwarding Detection)

TABLE 26 Default prioritized protocol table (Continued)

Protocol Packets
IPv6
OSPF / OSPF in 6to4
BGP / BGP in 6to4
RIPNG
MLD
ND6 / ND6 in 6to4
VRRP
VRRPEPIM / PIM in 6to4BFD (Bidirectional Forwarding Detection)
PIM / PIM in 6to4
Y.1731

Enhanced control packet prioritization

The Traffic Manager (TM) allows prioritization and scheduling of packets destined for the CPU to guarantee optimal control packet processing and to reduce protocol flapping. The TM achieves physical separation of CPU-bound data and control packets. The hierarchical structure supports four sets of eight priority queues. The four sets are as follows:

- Protocol set - Protocol packets that are prioritized by the network processor.
- Management set - Packets destined for the router; for example, Ping and Telnet.
- Flow set - Flow-driven packets to the CPU; for example, Unknown DA, DPA, Layer 2 broadcast and multicast, Multicast, VPLS SA Learning packets.
- Snoop set - CPU copy packets; for example, Regular Layer 2 SA learning, sFlow, ACL Logging, RPF Logging. The **rl-cpu-copy** command defines the rate shaping value for the snoop queues.

For each set of the eight priority queues, the priority queue 7 is given the highest rate, the priority queue 6 to queue 2 are given the same rate, and the priority queue 1 and queue 0 are given a lower rate. Each of the four sets of CPU queues is given equal weights. With the flow-driven packets given the same weight as the other packets, the protocol packets and flow-driven packets are queued separately so that the protocol packets are processed at a faster rate compared to the flow-driven packets.

The following table lists the protocol packets of the network processor prioritized protocol set based on their priorities.

TABLE 27 Network processor prioritized protocol packets

Priority categorization	Protocols
P7	LACP, UDLD (802.3ah), STP, RSTP, BPDU, VSRP, MRP, BFD, GRE-KA, IS-IS over GRE, G.8032, LLDP, non-CCM 802.1ag (Ethernet + MPLS Encapsulated), BFD (Single-hop, Multi-hop, and MPLS), IS-IS Hello, OSPFv2 Hello (GRE + Ethernet), OSPFv3 Hello (6to4 + Ethernet).

TABLE 27 Network processor prioritized protocol packets (Continued)

Priority categorization	Protocols
P6	IS-IS Non-Hello, OSPFv2 and OSPFv3 Non-Hello (Ethernet, GRE, 6to4), IPsec ESP Packets (for OSPFv3 over IPsec), OSPF, OSPF over GRE or 6to4, IS-IS, RIP, RIPNG, VRRP (Version 4 and Version 6), VRRP-E (Version 4 and Version 6).
P5	BGP, BGP over GRE or 6to4, PIM, PIM over GRE or 6to4, LDP (basic and extended), RSVP, CCP (MCT), 802.1ag CCM (Ethernet + MPLS Encapsulated).
P4	VPLS Encapsulated PIM, MSDP, MSDP over GRE, MSDP over VPLS.
P3	IGMP, VPLS Encapsulated IGMP, GRE Encapsulated IGMP, ARP, MLD, DHCP, BOOTP, ND6 and ND6 in 6to4.
P2	IPv4 Router Alert.
P1	New unassigned protocols.
P0	Existing unassigned protocols: GARP, L2-Trace.

Prioritizing management traffic to achieve QoS value

This feature introduces the ability to classify and prioritize the management traffic (SSH and Telnet) so that the outgoing management protocol packets from the device can receive a particular QoS value.

In congested networks, where management protocol packets may be dropped, or management protocol packets require a higher priority, the DSCP or IP precedence value from the incoming packets needs to be copied to the outgoing packets.

This feature is designed to change the per-hop behavior (PHB) of packets that are routed through the network. This will not affect the treatment of the packets inside the device.

Feature details

- CoS values from the received management protocol packet is copied to the transmittedpacket. It is implemented by using a CLI command **copy-received-cos** . The command is supported per application, irrespective of IP address, TCP or UDP port, etc.
- This feature supports IPv4 and IPv6.
- Each individual received packet is evaluated and its CoS value is copied to the transmitted packet. The priority changes during the flow, if the client changes the CoS values.
- This feature supports on SSH and Telnet.

Limitations and pre-requisites

If the DSCP encoding or PCP encoding is on, then the default encoding map must be used to achieve the expected behavior.

If user-defined Encode policy map is configured to mark the packets as they exit the device, then the DSCP or IP precedence value and PCP value may get changed based on the policy map and hence may not remain the same as the incoming packets.

Configuring the management traffic

Enable this feature by configuring the command for each management protocol.

This command is applied under global configuration.

By default, this feature will be disabled.

1. **telnet copy-received-cos** is for Telnet protocol.
2. **ip ssh copy-received-cos** is for SSH protocol.

```
device# telnet copy-received-cos
device# ip ssh copy-received-cos
```

For Telnet protocol, this the output of show command.

```
device #show telnet
Console connections:
        established, monitor enabled, privilege super-user
        you are connecting to this session
        1 seconds in idle
Telnet server status: Enabled
Telnet copy-received-cos status: Enabled
Telnet connections (inbound):
 1      closed
 2      closed
 3      closed
 4      closed
 5      closed
Telnet connections (outbound):
 6      closed
 7      closed
 8      closed
 9      closed
 10     closed
SSH server status: Enabled
SSH copy-received-cos status: Enabled
SSH connections (inbound):
 1      closed
 2      closed
 3      closed
 4      closed
 5      closed
 6      closed
 7      closed
 8      closed
 9      closed
 10     closed
 11     closed
 12     closed
 13     closed
 14     closed
 15     closed
 16     closed
SSH connections (outbound):
 17     closed
```

For SSH protocol, this the output of show command.

```
device #show ip ssh config
SSH server                   : Enabled
SSH port                     : tcp\22
Host Key                     : DSA 1024
Encryption                   : AES-256, AES-192, AES-128, 3-DES
Permit empty password        : No
Authentication methods       : Password, Public-key, Interactive
Authentication retries       : 3
Login timeout (seconds)      : 120
Idle timeout (minutes)       : 0
Strict management VRF        : Disabled
Copy Received CoS            : Enabled
SCP                          : Enabled
SSH IPv4 clients             : All
SSH IPv6 clients             : All
SSH IPv4 access-group        :
SSH IPv6 access-group        :
SSH Client Keys              : DSA(1024)
```

To disable the feature use these commands.

```
device# no telnet copy-received-cos
device# no ip ssh copy-received-cos
```

Configuring QoS

The QoS configuration process involves separate procedures for Ingress and Egress QoS Processing as described in the following major sections.

Configuring Ingress QoS procedures

The following procedures are required to configure a Brocade device for Ingress QoS processing:

- Creating Ingress Decode Policy Maps - If you want the priority and drop precedence values used within the device to be mapped to a specified value, you must create a Decode Priority map as described in Configuring Ingress decode policy maps on page 123.
- Binding Ingress Decode Policy Maps - If you want to apply an Ingress Policy Map other than the default, you must bind the Ingress Policy Map either globally or to a specified interface as described in Binding Ingress decode policy maps on page 128.
- Configuring a Force priority - Where there are multiple QoS values that can be used to determine the QoS level used on the device, the default policy is to determine the value used by performing a merge as described in Stage 2 Determine if a priority value should be forced or merged on page 107. Otherwise, you can specify a value that you want used from either the port or VLAN configured value or the DSCP, EXP or PCP values in the packets as described in Configuring a force priority on page 132.

Configuring Egress QoS procedures

The following procedures are required to configure a Brocade device for Egress QoS processing:

- Creating Egress Encode Policy Maps - If you want the priority and drop precedence values of packets leaving the device to be marked with a specified value, you must create an Encode Priority map as described in Configuring Egress encode policy maps on page 134.
- Binding Egress Encode Policy Maps - If you want to apply an Egress Policy Map other than the default, you must bind the Egress Policy Map either globally or to a specified interface as described in Binding an Egress encode EXP policy map on page 138.

Configuring QoS procedures applicable to Ingress and Egress

The following procedures are required to configure procedures applicable to both Ingress and Egress QoS Processing on a Brocade device:

- Enabling a Port to Use the DEI bit - You can configure the device to use the DEI bit when computing the drop precedence value for an incoming packet or encoding the DEI bit for transmitted frame as described in Enabling a port to use the DEI bit for Ingress and Egress processing on page 143.
- Specifying the Trust Level and Enabling Marking - If you want to use the **qos-tos trust** and **qos-tos mark** commands from pre-03.8.00 versions of the QoS feature, refer to Specifying the trust level and enabling marking on page 143.
- Support for Super Aggregate VLANS - If you want to use the enhanced QoS feature with Super Aggregate VLANs, refer to Configuring support for super aggregate VLANs on page 147.
- Support for QoS Configurations on LAG Ports - If you are configuring the enhanced QoS feature on ports within a LAG, refer to Configuring port-level QoS commands on LAG ports on page 147.

Configuring Ingress decode policy maps

Ingress Decode Policy Maps are created globally and are applied later either globally for all ports on a device or locally to specific port. To create an Ingress Decode Policy Map, you must first enter the QoS mapping configuration level of the command interface using the **qos-mapping** command, as shown in the following.

```
device(config)# qos-mapping
```

Configuration of each of the decode and encode mappings is described in the following sections:

- Configuring Ingress Decode DSCP Policy Maps
- Configuring Ingress Decode PCP Policy Maps
- Configuring Ingress Decode EXP Policy Maps

Configuring Ingress decode DSCP policy maps

The following procedures are used when configuring an Ingress Decode DSCP Policy Map:

- Naming an Ingress Decode DSCP Policy Map
- Configuring an Ingress Decode DSCP Policy Map

Naming an Ingress decode DSCP policy map

Once you are in the QoS configuration level, can define the name of a Ingress Decode DSCP Policy Map using the **pcp decode-map** command, as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# dscp decode-map Customer1
```

Syntax: [no] dscp decode-map *map-name*

The **no** option is used to delete a currently configured Ingress Decode DSCP Policy Map. If the Ingress Decode DSCP Policy Map is currently in use the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Ingress Decode DSCP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same *map-name* for different types of policy maps. For example, you can use the same name for an Ingress Decode DSCP Policy Map and a Ingress Decode EXP Policy Map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in Default QoS mappings on page 112.

Configuring an Ingress decode DSCP policy map

Once you have named an Ingress Decode DSCP Policy Map using the **dscp decode-map** command, you can set the values of the named Ingress Decode DSCP Policy Map. Setting the values in an Ingress Decode DSCP Policy Map involves specifying the value of the DSCP bits of an incoming packet and setting them to correspond to a value of 0 to 7 of the device’s internal priority. Optionally, you can set a drop precedence value of 0 to 3 in addition to the internal priority value.

To set the values of an Ingress Decode DSCP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **dscp-value** command as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# dscp decode-map Customer1
device(config-qos-mapping-dscp-decode)# dscp-value 32 to priority 5
drop-precedence 2
```

Syntax: [no] dscp-value *dscp-value* **to priority** *priority-value* [**drop-precedence** *dp-number*]

The *dscp-value* variable specifies the value of the DSCP bits within the packet header of the incoming packets. You can optionally specify multiple *dscp-value* variables if you want to specify more than one value to map to the same internal priority and drop precedence. Where DSCP values within a policy map are unspecified, the default mapping will be used.

The **priority** keyword together with the *priority-value* variable specifies the internal priority that the packets with the previously specified *dscp-value* value will be mapped to. The *priority-value* variable can be a value between 0 and 7. Please note, when generating the configuration file, a configured priority value that is the same as the value in the default priority map will not be shown.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you want to assign to incoming packets with the previously specified *dp-value* value. This value is specified in addition to a **priority***priority-value* value. The *dp-number* variable can be a value between 0 and 3. The default value is the value described in the default DSCP table. Please note, when generating the configuration file, a value for drop precedence will only be shown for non-default values.

When using the **no** option to negate a previously configured value of this command, observe the considerations described below.

1. You can negate both the **priority** and **drop-precedence** values (returning them to their default values) by using the **no** option with the original command only up to the **priority** value.

   For example: the following command has been used to set the map to assign an internal priority of "4" and a drop precedence of "2" to Ingress packets that have a DSCP value of "40".

   ```
   device(config-qos-mapping-dscp-decode)# dscp-value 40 to priority 4 drop-
   precedence 2
   ```

   To set the priority and **drop-precedence** values back to the default values, use the **no** option with the previous command up to where the **priority** value is configured, as shown in the following.

   ```
   device(config-qos-mapping-dscp-decode)# no dscp-value 40 to priority 4
   ```

After this command is executed, the **priority** and **drop-precedence** values for **dscp-value 40** will be returned to their default values as described in the default map tables that are defined in Default QoS mappings on page 112.

2. You can negate the **drop-precedence** value (returning it to its default value) without changing the currently configured **priority** value. This is done by using the **no** option with the original command that includes both the **priority** and **drop-precedence** values.

   For example: the following command has been used to set the priority map to assign an internal priority of "5" and a drop precedence of "1" to Ingress packets that have a DSCP value of "60".

   ```
   device(config-qos-mapping-dscp-decode)# dscp-value 60 to priority 5 drop-
   precedence 1
   ```

   To set the **drop-precedence** value back to the default value, use the **no** option with the previous command, as shown in the following.

   ```
   device(config-qos-mapping-dscp-decode)# no dscp-value 60 to priority 5 drop-
   precedence 1
   ```

   After this command is executed, the **priority** value will remain at 5 and the **drop-precedence** value will be returned to the default **drop-precedence** value for **dscp-value 60**, as described in the default map tables that are defined in Default QoS mappings on page 112.

Configuring Ingress decode PCP policy maps

The following procedures are used when configuring an Ingress Decode PCP Policy Map:

- Naming an Ingress Decode PCP Policy Map
- Configuring an Ingress Decode PCP Policy Map

Naming an Ingress decode PCP policy map

Once you are in the QoS configuration level, can define the name of an Ingress Decode PCP Policy Map using the **dscp decode-map** command, as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# pcp decode-map Customer1
```

Syntax: [no] pcp decode-map *map-name*

The **no** option is used to delete a currently configured Ingress decode PCP policy map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Ingress decode PCP policy map that you are defining. It can be up to 64 characters in length. You can specify the same map name for different types of maps. For example, you can use the same name for an Ingress decode PCP policy map and an Ingress decode DSCP policy map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in Default QoS mappings on page 112.

Configuring an Ingress decode PCP policy map

Once you have named an Ingress PCP Decode Policy Map using the **pcp decode-map** command, you can set the values of the named policy map. Setting the values in a policy map involves specifying the value of the PCP bits of an incoming packet and setting them to correspond to a value of 0 to 7 of the device's internal priority. Optionally, you can set a drop precedence value of 0 to 3 in addition to the internal priority value.

To set the values of an Ingress Decode PCP Policy Map, first specify name of the policy map and then populate the values in the Ingress Decode PCP Policy Map using the **pcp-value** command as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# pcp decode-map Customer1
device(config-qos-mapping-pcp-decode)# pcp-value 7 to priority 3 drop-precedence 2
```

Syntax: [no] pcp-value *pcp-value* [*pcp-value*] **to priority** *priority-value* [**drop-precedence** *dp-number*]

The *pcp-value* variable specifies the value of the PCP bits within the packet header of the incoming packets. You can optionally specify multiple *pcp-value* variables if you want to specify more than one value to map to the same internal priority and drop precedence. Where PCP values within a policy map are unspecified, the default mapping will be used.

The **priority** keyword together with the *priority-value* variable specifies the internal priority that the packets with the previously specified *pcp-value* value will be mapped to. The *priority-value* variable can be a value between 0 and 7. Please note, when generating the configuration file a configured priority value that is the same as the value in the default map will not be shown.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you want to assign to incoming packets with the previously specified *dp-value* value. This value is specified in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. The default value is 0. Please note, when generating the configuration file a value for drop precedence will only be shown for non-zero values.

When using the **no** option to negate a previously configured value of this command, observe the considerations described below.

1. You can negate both the **priority** and **drop-precedence** values (returning them to their default values) by using the **no** option with the original command only up to the **priority** value.

   For example: the following command has been used to set the map to assign an internal priority of "3" and a drop precedence of "2" to Ingress packets that have a PCP value of "7".

   ```
   device(config-qos-mapping-pcp-decode)# pcp-value 7 to priority 3 drop-precedence 2
   ```

   To set the **priority** and **drop-precedence** values back to the default values, use the **no** option with the previous command up to where the **priority** value is configured, as shown in the following.

   ```
   device(config-qos-mapping-pcp-decode)# no pcp-value 7 to priority 3
   ```

   After this command is executed, the **priority** and **drop-precedence** values for **pcp-value 7** will be returned to their default values as described in the default map tables that are defined in Default QoS mappings on page 112.
2. You can negate the **drop-precedence** value (returning it to its default value) without changing the currently configured **priority** value. This is done by using the **no** option with the original command that includes both the **priority** and **drop-precedence** values.

   For example: the following command has been used to set the priority map to assign an internal priority of "4" and a drop precedence of "2" to Ingress packets that have a PCP value of "6".

   ```
   device(config-qos-mapping-pcp-decode)# pcp-value 6 to priority 4 drop-precedence 2
   ```

To set the **drop-precedence** value back to the default value, use the **no** option with the previous command, as shown in the following.

```
device(config-qos-mapping-pcp-decode)# no pcp-value 6 to priority 4 drop-
precedence 2
```

After this command is executed, the **priority** value will remain at 4 and the **drop-precedence** value will be returned to the default **drop-precedence** value for **pcp-value 6** , as described in the default map tables that are defined in Default QoS mappings on page 112.

Configuring Ingress decode EXP Policy maps

The following procedures are used when configuring an Ingress Decode EXP Policy Map:

- Naming an Ingress Decode EXP Policy Map
- Configuring an Ingress Decode EXP Policy Map

Naming an Ingress decode EXP policy map

Once you are in the QoS configuration level, can define the name of a Ingress Decode EXP Policy Map using the **exp decode-map** command, as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# exp decode-map Customer1
```

Syntax: [no] exp decode-map *map-name*

The **no** option is used to delete a currently configured Ingress Decode EXP Policy Map. If the Ingress Decode EXP Policy Map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Ingress Decode EXP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same Ingress Decode EXP Policy Map for different types of policy maps. For example, you can use the same name for an Ingress Decode DSCP Policy Map and an Ingress Decode EXP Policy Map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in Default QoS mappings on page 112.

Configuring an Ingress decode EXP policy map

Once you have named an Ingress Decode EXP Policy Map using the **exp decode-map** command, you can set the values of the named policy map. Setting the values in a policy map involves specifying the value of the EXP bits of an incoming packet and setting them to correspond to a value of 0 to 7 of the device's internal priority value. Optionally, you can set a drop precedence value of 0 to 3 in addition to the internal priority value.

To set the values of an Ingress Decode EXP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **exp-value** command as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# exp decode-map Customer1
device(config-qos-mapping-exp-decode)# exp-value 7 to priority 5 drop-precedence 2
```

Syntax: [no] exp-value *exp-value* [*exp-value*] **to priority** *priority-value* [**drop-precedence** *dp-value*]

The *exp-value* variable specifies the value of the EXP bits within the packet header of the incoming packets. You can optionally specify multiple *exp-value* variables if you want to specify more than one value to map to the same internal priority and drop precedence values. Where EXP values within a policy map are unspecified, the default mapping will be used.

The **priority** keyword together with the *priority-value* variable specifies the internal priority value that the packets with the previously specified *exp-value* value will be mapped to. The *priority-value* variable can be a value between 0 and 7. Please note, when generating the configuration**drop-precedence** file a configured priority value that is the same as the value in the default priority map will not be shown.

The keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you want to assign to incoming packets with the previously specified *dp-value* value. This value is specified in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. The default value is the value described in the default EXP table. Please note, when generating the configuration file a value for drop precedence will only be shown for non-default values.

When using the **no** option to negate a previously configured value of this command, observe the considerations described below.

1. You can negate both the **priority** and **drop-precedence** values (returning them to their default values) by using the **no** option with the original command only up to the **priority** value.

   For example: the following command has been used to set the map to assign an internal priority of "5" and a drop precedence of "2" to Ingress packets that have an EXP value of "7".

   ```
   device(config-qos-mapping-exp-decode)# exp-value 7 to priority 5 drop-precedence 2
   ```

   To set the **priority** and **drop-precedence** values back to the default values, use the **no** option with the previous command up to where the **priority** value is configured, as shown in the following.

   ```
   device(config-qos-mapping-exp-decode)# no exp-value 7 to priority 5
   ```

   After this command is executed, the **priority** and **drop-precedence** values for **exp-value 7** will be returned to their default values as described in the default map tables that are defined in Default QoS mappings on page 112.
2. You can negate the **drop-precedence** value (returning it to its default value) without changing the currently configured **priority** value. This is done by using the **no** option with the original command that includes both the **priority** and **drop-precedence** values.

   For example: the following command has been used to set the priority map to assign an internal priority of "5" and a drop precedence of "2" to Ingress packets that have a EXP value of "7".

   ```
   device(config-qos-mapping-exp-decode)# exp-value 7 to priority 5 drop-precedence 2
   ```

   To set the **drop-precedence** value back to the default value, use the **no** option with the previous command, as shown in the following.

   ```
   device(config-qos-mapping-exp-decode)# no exp-value 7 to priority 5 drop-
   precedence 2
   ```

   After this command is executed, the **priority** value will remain at 5 and the **drop-precedence** value will be returned to the default **drop-precedence** value for **exp-value 7** , as described in the default map tables that are defined in Default QoS mappings on page 112.

Binding Ingress decode policy maps

You can bind an Ingress decode policy map globally or per-port using either the default policy map, an all zero policy map, or a user defined policy map. Additionally, for PCP, you can bind the following pre-defined policy maps: 7P1D, 6P2D, and 5P3D. The following procedures describe how to bind Ingress decode policy maps:

- Binding Ingress Decode DSCP Policy Maps
- Binding Ingress Decode PCP Policy Maps
- Binding Ingress Decode EXP Policy Maps

Binding Ingress decode DSCP policy maps

The following procedures describe how to configure the binding of Ingress Decode DSCP Policy Maps:

- Globally Binding an Ingress Decode DSCP Policy Map
- Binding an Ingress Decode DSCP Policy Map to a Port

Globally Binding an Ingress decode DSCP policy map

You can bind an Ingress Decode DSCP Policy Map globally for a Brocade device using the **qos dscp decode-policy** command as shown in the following.

```
device(config)# qos dscp decode-policy Customer1
```

Syntax: [no] qos dscp decode-policy *decode-map-name* {**default-map** | **all-zero-map**}

The *decode-map-name* variable is the name assigned to the Ingress Decode DSCP Policy Map that you want applied globally on the device. If you try to apply an *decode-map-name* value that has not been defined, the configuration will be rejected. If the *decode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Ingress Decode DSCP Policy Map globally on the device. Since the default Ingress Decode DSCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Ingress Decode DSCP Policy Map.

The **all-zero-map** option assigns a Ingress Decode DSCP Policy Map where all DSCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any DSCP information in the incoming packet.

Binding an Ingress decode DSCP policy map to a port

You can bind an Ingress Decode DSCP Policy Map to a specified port on a Brocade device using the **qos dscp encode-policy** command within an interface configuration, as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos dscp decode-policy Customer1
```

Syntax: [no] qos dscp decode-policy *decode-map-name* {**default-map** | **all-zero-map**}

The *decode-map-name* variable is the name assigned to the Ingress Decode DSCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Ingress Decode DSCP Policy Map to the port whose configuration this is under. Since the default Ingress Decode DSCP Policy Map is the global default setting, this option is only required when the device's global map has been set to a Ingress Decode DSCP Policy Map other than the default.

The **all-zero-map** option assigns an Ingress Decode DSCP Policy Map where all DSCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any DSCP information in the incoming packet

Binding an Ingress decode PCP policy map

The following procedures describe how to configure the binding of an Ingress Decode PCP Policy Map:

- Globally Binding an Ingress Decode PCP Policy Map
- Binding an Ingress Decode PCP Policy Map to a Port

Globally binding an Ingress decode PCP policy map

You can bind an Ingress Decode PCP Policy Map globally for a Brocade device using the **qos pcp decode-policy** command as shown in the following.

```
device(config)# qos pcp decode-policy Customer1
```

Syntax: [no] qos pcp decode-policy {*decode-map-name* | **default-map** | **all-zero-map** | **7P1D** | **6P2D** | **5P3**}

The *decode-map-name* variable is the name assigned to the Ingress Decode PCP Policy Map that you want applied globally on the device. If you try to apply an *decode-map-name* value that has not been defined, the configuration will be rejected. If the *decode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Ingress Decode PCP Policy Map globally on the device. The default policy map for PCP is the 8P0D decode map. Since the default Ingress Decode PCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Ingress Decode PCP Policy Map.

The **all-zero-map** option assigns an Ingress Decode PCP Policy Map where all PCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any PCP information in the incoming packet.

The **7P1D** option assigns the 7P1D Ingress Decode PCP Policy Map globally on the device.

The **6P2D** option assigns the 6P2D Ingress Decode PCP Policy Map globally on the device.

The **5P3D** option assigns the 5P3D Ingress Decode PCP Policy Map globally on the device.

NOTE
7P1D , **6P2D** and **5P3D** are as defined in the IEEE 802.1ad specification.

Binding an Ingress decode PCP policy map to a port

You can bind an Ingress Decode PCP Policy Map to a specified port on a Brocade device using the **qos pcp decode-policy** command as shown in the following.

```
device(config)# qos pcp decode-policy Customer1
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos pcp decode-policy Customer1
```

Syntax: [no] qos pcp decode-policy {*decode-map-name* | **default-map** | **all-zero-map** | **7P1D** | **6P2D** | **5P3D**}

The *decode-map-name* variable is the name assigned to the Ingress Decode PCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Ingress Decode PCP Policy Map to the port whose configuration this is under. Since the default Ingress Decode PCP Policy Map is the default setting, this option is only required when the device's global map has been set to an Ingress Decode PCP Policy Map other than the default.

The **all-zero-map** option assigns an Ingress Decode PCP Policy Map where all PCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any PCP information in the incoming packet.

The **7P1D** option assigns the 7P1D Ingress Decode PCP Policy Map to the port whose configuration this is under.

The **6P2D** option assigns the 6P2D Ingress Decode PCP Policy Map to the port whose configuration this is under.

The **5P3D** option assigns the 5P3D Ingress Decode PCP Policy Map to the port whose configuration this is under.

NOTE
7P1D , **6P2D** and **5P3D** are as defined in the IEEE 802.1ad specification.

Binding an Ingress decode EXP policy map

The following procedures describe how to configure the binding of an Ingress Decode EXP Policy Map:

- Globally Binding an Ingress Decode EXP Policy Map.
- Binding an Ingress Decode EXP Policy Map to a Port.

Globally binding an Ingress decode EXP policy map

You can bind an Ingress Decode EXP Policy Map globally for a Brocade device using the **qos exp decode-policy** command as shown in the following.

```
device(config)# qos exp decode-policy Customer1
```

Syntax: [no] qos exp decode-policy {*decode-map-name* | **default-map** | **all-zero-map**}

The *decode-map-name* variable is the name assigned to the Ingress Decode EXP Policy Map that you want applied globally on the device. If you try to apply a *decode-map-name* value that has not been defined, the configuration will be rejected. If the *decode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Ingress Decode EXP Policy Map globally on the device. Since the default Ingress Decode EXP Policy Map is the default setting, this option is only required when the device has been previously set to a different Ingress Decode EXP Policy Map.

The **all-zero-map** option assigns an Ingress Decode EXP Policy Map where all EXP values are mapped to priority 0 and drop precedence 0.

Binding an Ingress decode EXP policy map to a port

You can bind an Ingress Decode EXP Policy Map to a specified port on a Brocade device using the **qos exp decode-policy** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos exp decode-policy Customer1
```

Syntax: no qos exp decode-policy {*decode-map-name* | **default-map** | **all-zero-map**}

The *decode-map-name* variable is the name assigned to the Ingress Decode EXP Policy Map that you want applied to the port whose configuration this is under:

The **default-map** option assigns the default Ingress Decode EXP Policy Map to the port whose configuration this is under. Since the default Ingress Decode EXP Policy Map is the default setting, this option is only required when the device's global map has been set to an Ingress Decode EXP Policy Map other than the default.

The **all-zero-map** option assigns an Ingress Decode EXP Policy Map where all EXP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any EXP information in the incoming packet.

Configuring a force priority

In situations where there are conflicting priority values for packets on an Ingress port, that conflict can be resolved by performing a priority merge or by using a **force** command to direct the device to use a particular value above other values. A **force** command can be configured for each of the following:

- Force to the values configured on a port
- Force to the value configured for a VLAN
- Force to the value in the DSCP bits
- Force to the value in the EXP bits
- Force to the value in the PCP bits
- Force to a value specified within an ACL

Configuring a force priority for a port

You can configure an Ingress port with a priority to apply to packets that arrive on it using the **priority** command.

To configure an Ingress port with a priority, use the **priority** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)priority 6
```

Syntax: [no] priority *priority-value*

The *priority-value* variable is a value between 0 and 7. The default value is 0.

Once a port has been configured with a priority using the **priority** command, you can then configure the port (using the **priority force** command) to force the configured priority when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the port-configured priority, use the **priority force** command as shown in the following:

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)priority force
```

Syntax:[no] priority force

Configuring a force drop precedence for a port

You can configure an Ingress port with a drop precedence to apply to packets that arrive on it using the **drop-precedence** command.

To configure an Ingress port with a drop precedence, use the **drop-precedence** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)drop-precedence 3
```

Syntax: [no] drop-precedence *dp-value*

The *dp-value* variable is a value between 0 and 3.

Once a port has been configured with a drop precedence using the **drop-precedence** command, you can then configure the port (using the **drop-precedence force** command) to force the configured drop precedence when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the port-configured drop precedence, use the **drop-precedence force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)drop-precedence force
```

Syntax: [no] drop-precedence force

Configuring a force priority for a VLAN

By default, VLANs have priority 0. To change a port-based VLAN’s QoS priority, use the following method. The priority applies to outbound traffic on ports in the VLAN.

To change the QoS priority of port-based VLAN 20 on a Chassis device to priority queue 7, enter the following commands.

```
device(config)# vlan 20
device(config-vlan-20)# priority 7
```

Syntax: [no] priority *num*

The *num* parameter can be from 0 - 7 and specifies one of the eight QoS queues.

NOTE
When you apply the VLAN priority command with running traffic, it may drop packets for a short period of time. This is normal.

Once a VLAN has been configured with a priority using the **priority** command, you can then configure the VLAN (using the **priority force** command) to force the configured priority when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the VLAN-configured priority, use the **priority force** command as shown in the following.

```
device(config)# vlan 20
device(config-vlan-20) priority force
```

Syntax: [no] priority force

Configuring force priority to the DSCP value

You can configure an Ingress port (using the **qos dscp force** command) to force the configured DSCP value when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the DSCP value, use the **qos dscp force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos dscp force
```

Syntax: [no] qos dscp force

Configuring force priority to the EXP value

You can configure an Ingress port (using the **qos exp force** command) to force the configured EXP value when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the EXP value, use the **qos exp force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos exp force
```

Syntax: qos exp force

Configuring force priority to the PCP value

You can configure an Ingress port (using the **qos pcp force** command) to force the configured PCP value when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the PCP value, use the **qos pcp force** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos pcp force
```

Syntax: qos pcp force

Configuring force priority to a value specified by an ACL

You can use the **priority-force** keyword within an ACL to apply a priority to specified traffic as described in "Filtering and priority manipulation based on 802.1p priority" in the *Multi-Service IronWare Security Configuration Guide* .

Configuring Egress encode policy maps

Egress Encode Policy Maps are created globally and are applied later either globally for all ports on a device or locally to specific port. To create an Egress Encode Policy Map, you must first enter the QoS mapping configuration level of the command interface using the **qos-mapping** command, as shown in the following.

```
device(config)# qos-mapping
```

Configuration of each of the Egress Encode Policy Maps is described in the following sections:

- Configuring Egress Encode DSCP Policy Maps
- Configuring Egress Encode PCP Policy Maps
- Configuring Egress Encode EXP Policy Maps

Configuring Egress encode DSCP policy maps

The following procedures are used when configuring an Egress Encode DSCP Policy Map:

- Naming an Egress Encode DSCP Policy Map
- Configuring an Egress Encode DSCP Policy Map

Naming an Egress encode DSCP policy map

Once you are in the QoS configuration level, can define the name of an Egress Encode DSCP Policy Map using the **dscp encode-map** command, as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# dscp encode-map Customer1
```

Syntax: [no] dscp encode-map *map-name*

The **no** option is used to delete a currently configured Egress Encode DSCP Policy Map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Egress Encode DSCP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same policy map name for different types of maps. For example, you can use the same policy pam name for an Egress Encode DSCP Policy Map and an Egress Encode EXP Policy map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in Default QoS mappings on page 112.

Configuring an Egress encode DSCP policy map

Once you have named an Egress Encode DSCP Policy Map using the **dscp encode-map** command, you can set the values of the named encode policy map. Setting the values in an Egress Encode DSCP Policy Map involves specifying a DSCP value to be marked in outgoing packets for a specified priority value (0 - 7) and optionally a drop precedence value (0 - 3).

To set the values of an Egress Encode DSCP Policy Map, first specify name of the policy map and then populate the values in the Egress Encode DSCP Policy Map using the **priority** command as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# dscp encode-map Customer1
device(config-qos-mapping-dscp-encode)# priority 7 drop-precedence 2 to
dscp-value 3
```

Syntax: [no] priority *priority-value* [**drop-precedence** *dp-value*] [*dp-value*] **to dscp-value** *dscp-value*

The **priority** keyword together with the *priority-value* variable specifies the internal priority value that egress packets will be marked from. The *priority-value* variable can be a value between 0 and 7. For unspecified priority values, the default mapping values are used.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you specify in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. Multiple *dp-number* variables can be configured in a single command. The default value is "any" which means that drop priorities 0 - 3 are assigned. If a drop precedence value is specified only for a subset of values, the entries with unspecified values will be initialized as specified in the default mapping.

The *dscp-value* variable specifies the value that will be marked onto the DSCP bits within the packet header of the outgoing packets. This applies to packets that match the **priority** *priority-value* and **drop-precedence** values specified in this command.

The **no** option allows you to negate a previously configured value and return to the default mapping for the specified **priority** *priority-value* and **drop-precedence** values.

Configuring an Egress encode PCP policy map

The following procedures are used when configuring an Egress Encode PCP Policy Maps:

- Naming an Egress Encode PCP Policy Map
- Configuring an Egress Encode PCP Policy Map

Naming an Egress encode PCP policy map

Once you are in the QoS configuration level, can define the name of an Egress Encode PCP Policy Map using the **pcp encode-map** command, as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# pcp encode-map Customer1
```

Syntax: [no] pcp encode-map *map-name*

The **no** option is used to delete a currently configured Egress Encode PCP Policy Map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Egress Encode PCP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same policy map name for different types of policy maps. For example, you can use the same name for an Egress Encode PCP Policy Map and an Egress Encode EXP Policy Map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in Default QoS mappings on page 112.

Configuring an Egress encode PCP policy map

Once you have named an Egress Encode PCP Policy Map using the **pcp encode-map** command, you can set the values of the named policy map. Setting the values in an Egress Encode PCP Policy Map involves specifying a PCP value to be marked in outgoing packets for a specified internal priority value (0 - 7) and optionally a drop precedence value (0 - 3).

To set the values of an Egress Encode PCP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **priority** command as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# pcp encode-map Customer1
device(config-qos-mapping-pcp-encode)# priority 7 drop-precedence 2 to pcp-value 3
```

Syntax: [no] priority *priority-value* [**drop-precedence** *dp-value*] [*dp-value*] **to pcp-value** *pcp-value*

The **priority** keyword together with the *priority-value* variable specifies the priority value that the egress packets will be marked with. The *priority-value* variable can be a value between 0 and 7. For unspecified priority values, the default mapping values are used.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you specify in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. Multiple *dp-number* variables can be configured in a single command. The default value is "any" which means that drop priorities 0 - 3 are assigned. If a drop precedence value is specified only for a subset of values, the entries with unspecified values will be initialized as specified in the default mapping.

The *pcp-value* variable specifies the value that will be marked onto the PCP bits within the packet header of the outgoing packets. This applies to packets that match the **priority** and **drop-precedence** values specified in this command.

The **no** option allows you to negate a previously configured value and return to the default mapping for the specified **priority** *priority-value* and **drop-precedence** values.

Configuring an Egress Encode EXP policy map

The following procedures are used when configuring an Egress Encode EXP Policy Map:

- Naming an Egress Encode EXP Policy Map
- Configuring an Egress Encode EXP Policy Map

Naming an Egress encode EXP policy map

Once you are in the QoS configuration level, can define the name of an Egress Encode EXP Policy Map using the **exp encode-map** command, as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# exp encode-map Customer1
```

Syntax: [no] exp encode-map *map-name*

The *map-name* variable specifies the name of the Egress Encode EXP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same policy map name for different types of policy maps. For example, you can use the same name for an Egress Encode EXP Policy and an Egress Encode DSCP Policy Map.

The **no** option is used to delete a currently configure Egress Encode EXP Policy Map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in Default QoS mappings on page 112.

Configuring an Egress encode EXP policy map

Once you have named an Egress Encode EXP Policy Map using the **exp encode-map** command, you can set the values of the named encode policy map. Setting the values in an Egress Encode EXP Policy Map involves specifying an EXP value to be marked in outgoing packets for a specified priority value (0 - 7) and optionally a drop precedence value (0 - 3).

To set the values of an Egress Encode EXP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **priority** command as shown in the following.

```
device(config)# qos-mapping
device(config-qos-mapping)# exp encode-map Customer1
device(config-qos-mapping-exp-encode)# priority 7 drop-precedence 2 to exp-value 3
```

Syntax: [no] priority *priority-value* [**drop-precedence** *dp-value*] [*dp-value*] **to exp-value** *exp-value*

The **priority** keyword together with the *priority-value* variable specifies the internal forwarding value of the egress packets. The *priority-value* variable can be a value between 0 and 7. For unspecified priority values, the default mapping values are used.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you specify in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. Multiple *dp-number* variables can be configured in a single command. The default value is "any" which means that drop priorities 0 - 3 are assigned. If a drop precedence value is specified only for a subset of values, the entries with unspecified values will be initialized as specified in the default mapping.

The *exp-value* variable specifies the value that will be marked onto the EXP bits within the packet header of the outgoing packets. This applies to packets that match the no and **drop-precedence** values specified in this command.

The **no** option allows you to negate a previously configured value and return to the default mapping for the specified **priority** and **drop-precedence** values.

Binding an Egress encode EXP policy map

You can bind an Egress Encode Policy map globally or per-port using either the default policy map, an all zero policy map, or a user defined policy map. Additionally, for PCP, you can bind the following pre-defined policy maps: 7P1D, 6P2D, and 5P3D. The following procedures describe how to bind Egress Encode Policy Maps:

- Binding an Egress Encode DSCP Policy Map
- Binding an Egress Encode PCP Policy Map
- Binding an Ingress Encode EXP Policy Map

Binding an Egress encode DSCP policy map

The following procedures describe how to configure the binding of an Egress Encode DSCP Policy Map:

- Globally Binding an Egress Encode DSCP Policy Map
- Binding an Egress Encode DSCP Policy Map to a Port

Globally binding an Egress encode DSCP policy map

You can bind an Egress Encode DSCP Policy Map globally for a Brocade device using the **qos dscp encode-policy** command as shown in the following.

```
device(config)# qos dscp encode-policy Customer1
```

Syntax: [no] qos dscp encode-policy *encode-map-name* {**default-map** | **all-zero-map**}

The *encode-map-name* variable is the name assigned to the Egress Encode DSCP Policy Map that you want applied globally on the device. If you try to apply a *encode-map-name* value that has not been defined, the configuration will be rejected. If the *encode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Egress Encode DSCP Policy Map globally on the device. Since the default Egress Encode DSCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Egress Encode DSCP Policy Map. When configured globally, the **qos dscp encode-policy default-map** command will not be displayed within the configuration even if it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode DSCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos dscp encode-policy** command is not configured, then the **no qos pcp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos dscp encode-policy default-map** command is not configured, then the **no qos dscp encode-policy default-map** command will still be allowed because the **qos dscp encode-policy default-map** is the default configuration.

Binding an Egress encode DSCP policy map to a port

You can bind an Egress Encode DSCP Policy Map to a specified port on a Brocade device using the **qos dscp encode-policy** command within an interface configuration, as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos dscp encode-policy Customer1
device(config-if-e10000-10/1)qos dscp encode-policy on
```

Syntax: [no] qos dscp encode-policy *encode-map-name* {**default-map** | **all-zero-map**}

NOTE
The **qos dscp encode-policy on** command is shown in this example because unlike PCP or EXP, the DSCP encode policy is off by default.

The *encode-map-name* variable is the name assigned to the Egress Encode DSCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Egress Encode DSCP Policy Map to the port whose configuration this is under. Since the default Egress Encode DSCP Policy Map is the global default setting, this option is only required when the device's global map has been set to an Egress Encode DSCP Policy Map other than the default. The **qos dscp encode-policy** command will not be displayed within the configuration unless it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode DSCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos pcp encode-policy** command is not configured, then the **no qos pcp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured Egress Encode DSCP Policy Map. If the **qos dscp encode-policy default-map** command is not configured, then the **no qos dscp encode-policy default-map** command will generate an error message because the **qos dscp encode-policy default-map** command was never configured on the port.

NOTE
The Egress Encode DSCP Policy is applied to the egress port, allowing different egress policies to be configured on each port.

Enabling and disabling an Egress Encode DSCP Policy Map on a port

To enable or disable an Egress Encode DSCP Policy Map on a port, use the **qos dscp encode-policy** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos dscp encode-policy on
```

Syntax: [no] qos dscp encode-policy {**on** | **off**}

The **on** option enables DSCP encode on the port. The **qos dscp encode-policy on** command will not be displayed within the configuration unless it is explicitly configured.

The **off** option disables DSCP encode on the port. This is the default setting.

NOTE
Enable Encode DSCP Policy Map on the corresponding mirror port as well by using the **qos dscp encode-policy on** command.

Binding Egress encode PCP policy map

The following procedures describe how to configure the binding of an Egress Encode PCP Policy Map:

- Globally Binding an Egress Encode PCP Policy Map Policy
- Binding an Egress Encode PCP Policy Map to a Port

Globally binding an Egress Encode PCP Policy Map

You can bind an Egress Encode PCP Policy Map globally for a Brocade device using the **qos pcp encode-policy** command as shown in the following.

```
device(config)# qos pcp encode-policy Customer1
```

Syntax: [no] qos pcp encode-policy {*encode-map-name* | **default-map** | **all-zero-map** | **7P1D** | **6P2D** | **5P3D**}

The *encode-map-name* variable is the name assigned to the Egress Encode PCP Policy Map that you want applied globally on the device. If you try to apply a *encode-map-name* value that has not been defined, the configuration will be rejected. If the *encode-map-name* value has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Egress Encode PCP Policy Map globally on the device. Since the default Egress Encode PCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Egress Encode PCP Policy Map. When configured globally, the **qos pcp encode-policy default-map** command will not be displayed within the configuration even if it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode PCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **7P1D** option assigns the 7P1D Egress Encode PCP Policy Map globally on the device.

The **6P2D** option assigns the 6P2D Egress Encode PCP Policy Map globally on the device.

The **5P3D** option assigns the 5P3D Egress Encode PCP Policy Map globally on the device.

NOTE
7P1D, **6P2D** and **5P3D** are as defined in the IEEE 802.1ad specification.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos pcp encode-policy default-map** command is not configured, then the **no qos pcp encode-policy default-map** command will still be allowed because the **qos pcp encode-policy default-map** is the default configuration.

Binding an Egress encode PCP policy map to a port

You can bind an Egress Encode PCP Policy Map to a specified port on a Brocade device using the **qos pcp encode-policy** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos pcp encode-policy Customer1
```

Syntax: [no] qos pcp encode-policy {*encode-map-name* | **default-map** | **all-zero-map** | **7P1D** | **6P2D** | **5P3D**}

The *encode-map-name* variable is the name assigned to the Egress Encode PCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Egress Encode PCP Policy Map to the port whose configuration this is under. Since the default Egress Encode PCP Policy Map is the default setting, this option is only required when the device's global map has been set to an Egress Encode PCP Policy Map other than the default. The **qos pcp encode-policy default-map** command will not be displayed within the configuration unless it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode PCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **7P1D** option assigns the 7P1D Egress Encode PCP Policy Map to the port whose configuration this is under.

The **6P2D** option assigns the 6P2D Egress Encode PCP Policy Map to the port whose configuration this is under.

The **5P3D** option assigns the 5P3D Egress Encode PCP Policy Map to the port whose configuration this is under.

NOTE
7P1D, **6P2D** and **5P3D** are as defined in the IEEE 802.1ad specification.

The **no** option allows you to withdraw a previously configured Egress Encode PCP Policy Map. If the **qos pcp encode-policy** command is not configured, then the **no qos pcp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured Egress Encode PCP Policy Map. If the **qos pcp encode-policy default-map** command is not configured, then the **no qos pcp encode-policy default-map** command will generate an error message because the **qos pcp encode-policy default-map** command was never configured on the port.

Enabling and disabling an Egress Encode PCP Policy Map on a port

To enable or disable an Egress Encode DSCP Policy Map on a port, use the **qos pcp encode-policy** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos pcp encode-policy on
```

Syntax: qos pcp encode-policy {**on** | **off**}

The **on** option enables PCP encode-policy on the port. This is the default setting. The **qos pcp encode-policy on** command will not be displayed within the configuration unless it is explicitly configured.

The **off** option disables PCP encode-policy on the port.

Binding Egress encode EXP policy maps

The following procedures describe how to configure the binding of an Egress Encode EXP Policy Map:

- Globally Binding an Egress Encode EXP Policy Map
- Binding an Egress Encode EXP Policy Map to a Port

Globally binding an Egress Encode EXP Policy Map

You can bind an Egress Encode EXP Policy Map globally for a Brocade device using the **qos exp encode-policy** command as shown in the following.

```
device(config)# qos exp encode-policy Customer1
```

Syntax: [no] qos exp encode-policy {*encode-map-name* | **default-map** | **all-zero-map**}

The *encode-map-name* variable is the name assigned to the Egress Encode EXP Policy Map that you want applied globally on the device. If you try to apply an *encode-map-name* value that has not been defined, the configuration will be rejected. If the *encode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Egress Encode EXP Policy Map globally on the device. Since the default Egress Encode EXP Policy Map is the default setting, this option is only required when the device has been previously set to a different Egress Encode EXP Policy Map. When configured globally, the **qos exp encode-policy default-map** command will not be displayed within the configuration even if it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode EXP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured Egress Encode EXP Policy Map. If the **qos exp encode-policy** command is not configured, then the **no qos exp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured Egress Encode EXP Policy Map. If the **qos exp encode-policy default-map** command is not configured, the **no qos exp encode-policy default-map** command will still be allowed because **qos exp encode-policy default-map** is the default configuration.

Binding an Egress Encode EXP Policy Map to a port

You can bind an Egress Encode EXP Policy Map to a specified port on a Brocade device using the **qos exp encode-policy** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos exp encode-policy Customer1
```

Syntax: [no] qos exp encode-policy {*encode-map-name* | **default-map** | **all-zero-map**}

The *encode-map-name* variable is the name assigned to the Egress Encode EXP Policy Map that you want applied to the port whose configuration this is under:

The **default-map** option assigns the default Egress Encode EXP Policy Map to the port whose configuration this is under. Since the default Egress Encode EXP Policy Map is the default setting, this option is only required when the device's global policy map has been set to an Egress Encode EXP Policy Map other than the default. The **qos exp encode-policy default-map** command will not be displayed within the configuration unless it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode EXP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos exp encode-policy default-map** command is not configured, then the **no qos exp encode-policy default-map** command will generate an error message because the **qos exp encode-policy default-map** command was never configured on the port.

Enabling and disabling an Egress Encode EXP Policy Map on a port

To enable or disable an Egress Encode EXP Policy Map on a port, use the **qos exp encode-policy** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos exp encode-policy on
```

Syntax: [no] qos exp encode-policy {**on** | **off**}

The **on** option enables EXP encode on the port. This is the default setting. The **qos exp encode-policy on** command will not be displayed within the configuration unless it is explicitly configured.

The **off** option disables EXP encode on the port.

Enabling a port to use the DEI bit for Ingress and Egress processing

In the IEEE 802.1ad specification, two types of tag are defined:

- Customer VLAN tag (C-TAG)
- Service VLAN tag (S-TAG)

The semantics and structure of the S-TAG is identical to that of the C-TAG, with the exception that bit 5 in octet 1, the Drop Eligible Indicator (DEI) bit, dis used to indicate if the packet is drop eligible. This allows all 3 bits in the PCP ID to be used for indicating priority of the packet with the drop precedence indicated by the DEI bit. The IEEE 802.1ad requires that if this capability is provided, it must be independently manageable for each port.

On the Brocade device the **qos use-dei** command can be configured at the port level to allow a drop-precedence value for incoming packet to be computed based on the DEI bit. Additionally, if this command is configured, then a drop-eligible parameter will be encoded in the DEI bit of transmitted frames. If the internal drop precedence of the packet is 2 or 3, the DEI will be transmitted as 1; otherwise it will be transmitted as 0.

This command is configured as described in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos use-dei
```

Syntax: no qos use-dei

NOTE
This command applies for both Ingress and Egress processing.

Specifying the trust level and enabling marking

The following commands were retained from pre-03-8.00 versions of Multi-Service IronWare software:

- qos-tos trust
- qos-tos mark

These commands operate on the QoS values within the packets as they arrive on the device. The **qos-tos trust** command specifies which value among the following to use to classify the packet for marking: **cos**, **ip-prec**, and **dscp**. The **qos-tos mark** command specifies a CoS or DSCP value to mark on outgoing packets as specified by the mappings described in Packet mapping commands on page 145.

NOTE
You cannot use these commands and other L4 features such as:

- IPv4 ACLs and IPv4 ACL-based rate-limiting
- Layer 2 ACLs and Layer 2 ACL-based rate-limiting
- PBR
- VLAN ID and Inner VLAN ID translation on the same interface

NOTE
The design of this feature requires that the **qos-tos trust** and **qos-tos mark** commands be used together.

NOTE
In versions of the Multi-Service IronWareprior to 03.8.00, before configuring the **qos-tos trust** and **qos-tos** mark commands, you had to configure the **port-priority** command at global CONFIG level. Beginning with version 03.8.00, the **port-priority** command is no longer supported. You can now directly configure the **qos-tos trust** and **qos-tos mark** commands at the interface-level. However without the **port-priority** command configured, the per-port DSCP decode map is not initialized as previously. For information concerning the upgrade of a previously configured DSCP decode map, refer to DSCP-priority mapping commands on page 111.

Specifying the trust level

The trust level specifies where you want the device to get the QoS value for a packet received on the interface.

To set the trust level for an interface to IP Precedence, enter the following command at the configuration level for the interface.

```
device(config-if-1/1)# qos-tos trust ip-prec
```

Syntax: [no] qos-tos trust {**cos** | **ip-prec** | **dscp**}

The **cos | ip-prec | dscp** parameter specifies the trust level:

- **cos** - The device uses the IEEE 802.1p (CoS) priority value in the packet's Ethernet frame header. Use this trust option when you plan to mark the packet's DSCP value based on the incoming IEEE 802.1p value.
- **ip-prec** - The device uses the three most-significant bits in the packet's ToS field and interprets them as an IP precedence value. Use this trust option when the incoming packet is from a device that does not support DSCP and you need to mark the packet for QoS on DSCP devices.
- **dscp** - The device uses the six most-significant bits in the packet's ToS field and interprets them as a DSCP value.

Enabling marking

Marking changes the value of an outbound packet's IEEE 802.1p priority field, DSCP field, or both to match the results of the QoS mappings performed by the device. When you enable marking on an interface, the marking applies to packets that enter the device through that interface.

The following example enables marking for traffic that arrives on port 1/1 and enables the **qos pcp encode-policy on** command on egress port 1/14, as shown.

```
device(config-if-e10000-1/1)# qos-tos mark cos
device(config-if-e10000-1/1)# interface ethernet 1/14
device(config-if-e10000-1/1)# qos pcp encode-policy on
```

This command enables marking of the IEEE 802.1p field in the Ethernet frame.

Syntax: [no] qos-tos mark {**cos** | **dscp**}

The **cos** and **dscp** parameters are exclusive and specify the type of marking:

- **cos** - The device changes the outbound packet's IEEE 802.1p priority value to match the results of the device's QoS mapping from the specified trust level.
- **dscp** - The device changes the outbound packet's IEEE 802.1p priority value to match the results of the device's QoS mapping from the specified trust level.

NOTE
In release 03.8.00 and later, the **qos pcp encode-policy on** command must be configured when the **qos-tos mark cos** command is configured. The **qos pcp encode-policy** command is on by default and does not require explicit configuration unless it has been configured to be **off** .

NOTE
You can't apply an ACL to an interface in the outbound direction to change the priority of certain types of traffic.

Packet mapping commands

The **qos-tos trust** command, that is retained from pre-03.8.00 versions of Multi-Service IronWare software, described in the proceeding section, specifies that a COS, IP-Precedence, or DSCP value received on an Ingress port will be used to determine the QoS value that is marked on an outgoing packet on an egress port. The **qos-tos mark** command that is also retained from pre-03.8.00 versions, directs the device to mark outgoing packets with a COS or DSCP value as specified in the command. The value to be marked is determined by a mapping between the value received on the Ingress port and another value that you set using one of the following procedures:

- Changing the CoS -> DSCP Mappings
- Changing the IP Precedence -> DSCP Mappings
- Changing the DSCP -> DSCP Mappings

Changing the CoS - DSCP mappings

The CoS -> DSCP mappings are used if the trust level is CoS as set by the **qos-tos trust** command.

To change the CoS -> DSCP mappings, enter commands such as the following at the global CONFIG level of the CLI.

```
device(config)# qos-tos map cos-dscp 0 33 25 49 17 7 55 41
device(config)# ip rebind-acl all
```

This command configures the mappings displayed in the COS-DSCP map portion of the QoS information display.

```
device(config-if-1/1)# show qos-tos
```

...portions of table omitted for simplicity...

```
COS-DSCP map:
     COS: 0 1 2 3 4 5 6 7
  ---------------------------------------------------------
     dscp: 0 33 25 49 17 7 55 41
```

Syntax: [no] qos-tos cos-dscp *dscp1 dscp2 dscp3 dscp4 dscp5 dscp6 dscp7 dscp8*

The *dscp1 ... dscp8* parameters specify the DSCP values you are mapping to the eight CoS values. You must enter DSCP values for all eight CoS values, in order from CoS value 0 - 7.

NOTE
To place a qos-tos mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change. This applies to mappings that are configured using the **qos-tos map** command.

Changing the IP precedence - DSCP mappings

The IP precedence -> DSCP mappings are used if the trust level is IP Precedence as set by the **qos-tos trust** command.

To change the IP precedence -> DSCP mappings, enter commands such as the following at the global CONFIG level of the CLI.

```
device(config)# qos-tos map ip-prec-dscp 0 32 24 48 16 8 56 40
device(config)# ip rebind-acl all
```

This command configures the mappings displayed in the IP Precedence-DSCP map portion of the QoS information display.

```
device(config-if-1/1)# show qos-tos
```

...portions of table omitted for simplicity...

```
IP Precedence-DSCP map:
  ip-prec:     0     1     2     3     4     5     6     7
  ---------------------------------------------------------
     dscp:     0    32    24    48    16     8    56    40
```

For information about the rest of this display, refer to Displaying QoS configuration information on page 149.

Syntax: no qos-tos map ip-prec-dscp *dscp1 dscp2 dscp3 dscp4 dscp5 dscp6 dscp7dscp8*

The *dscp1 ... dscp8* parameters specify the DSCP values you are mapping to the IP precedence values. You must enter DSCP values for all eight IP precedence values, in order from IP precedence value 0 - 7.

NOTE
To place a qos-tos mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change. This applies to mappings that are configured using the **qos-tos map** command.

Changing the DSCP - DSCP mappings

To change a DSCP -> DSCP mapping, enter a command such as the following at the global CONFIG CLI level.

```
device(config)# qos-tos map dscp-dscp 0 10
device(config)# ip rebind-acl all
```

This command changes the mapping of DSCP value 0 from 0 to 10.

Syntax: [no] qos-tos map dscp-dscp *old-dscp-value* [*old-dscp-value...*] **to** *new-dscp-value* [*new-dscp-value...*]

You can change up to eight DSCP values in the same commend. Make sure you enter the old values and their new values in the same order.

NOTE
To place a qos-tos mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change. This applies to mappings that are configured using the **qos-tos map** command.

Configuring support for super aggregate VLANs

In a super-aggregate VLAN application, you can optionally configure an untagged interface to copy the QOS bits from the tag value set by the edge device to the tag value set by the core device. This is only supported if the incoming packet has ETYPE 0x8100. This can be configured using the **qos decode-cvlan-pcp** command as shown in the following.

```
device(config)# interface ethernet 10/1
device(config-if-e10000-10/1)qos decode-cvlan-pcp
```

Syntax: [no] qos decode-cvlan-pcp

NOTE
The command **aggregated-vlan-copy-cos** is available at the physical interface level to copy the COS value from the internal to the external VLAN tag (for SAV). This command will be automatically migrated to the new command **qos decode-cvlan-pcp** .

Configuring port-level QoS commands on LAG ports

When applying port-level QoS commands to ports in a LAG, the rules can differ according the following:

- For port-level QoS Configurations where QoS Values are Applied Directly to the Port. These commands include the following: **priority**, **priority force**, **drop-precedence**, **drop-precedence force**.
- For Port-level QoS configurations using commands that begin with the **qos** keyword. These commands include: **qos use-dei,qos dscp decode-policy**, **qos pcp decode-policy**, **qos exp decode-policy**, **qos dscp force**, **qos pcp force**, **qos exp force**, **qos dscp encode-policy**, **qos pcp encode-policy**, and **qos exp encode-policy**.

LAG configuration rules where QoS values are applied directly to the port

In port-level QoS Configurations where QoS values are applied directly to the port, the considerations listed below must be followed.

1. Each port that is configured into the LAG, must have the same **priority** , **priority force** , **drop-precedence** , and **drop-precedence force** configuration.

   If you try to configure a LAG with ports that have a different configuration for these commands, the LAG deployment will fail and you will get an error message as shown in the following.

   ```
   device(config)# lag mylag static
   device(config-lag-mylag)# ports eth 10/1 to 10/2
   device(config-lag-mylag)# primary 10/1
   device(config-lag-mylag)# deploy
   port 10/1 priority is 5, but port 10/2 priority is 0
   Error: port 10/1 and port 10/2 have different configurations
   LAG mylag deployment failed!
   device(config-lag-mylag)#
   ```
2. If you have already formed a LAG with the same configuration, you can change the configuration by making changes to the LAG's primary port.
3. If the LAG configuration is deleted, each of the port in the LAG (primary and secondary) will inherit the QoS configuration of the primary port.

LAG configuration rules for QoS configurations using commands that begin with the qos keyword

In port-level QoS Configurations where QoS Configurations Using Commands that begin with the **qos** keyword are used, the considerations listed below must be followed.

1. The secondary ports configured in the LAG must not have any QoS values configured on them.
2. The **qos** commands that are configured on the primary port are applied to all ports in the LAG.
3. After the LAG is formed, you can change the QoS configuration for all ports in the LAG by making changes to the LAG's primary port, but you cannot change the QoS configurations directly on any of the secondary ports.
4. If the LAG is deleted, the QoS configuration will be retained on the primary and secondary ports.

Displaying QoS information

You can display the following QoS information as described:

- **QoS Configuration Information** - Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can display the priority and drop-precedence values mapped between values internal to the device and values that are received at the device or marked on packets leaving the device. This is described in Displaying QoS configuration information on page 149.

- **QoS Packet and Byte Statistics** - Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can enable and display the contents of the QoS Packet and Byte Counters as described in Displaying QoS packet and byte counters on page 151.

Displaying QoS configuration information

You can display the following QoS Configuration information:

- QoS Decode Policy Map Configurations
- QoS Policy Map Binding Configurations

Displaying QoS Decode Policy Map configurations

To display QoS Decode Policy Map configuration information, enter the following command at any level of the CLI.

```
device(config)# show qos-map dscp decode-map test1
DSCP decode map test1
  DSCP  0 to priority  0 drop-precedence  0
  DSCP  1 to priority  0 drop-precedence  0
  DSCP  2 to priority  0 drop-precedence  1
  DSCP  3 to priority  0 drop-precedence  1
  DSCP  4 to priority  0 drop-precedence  2
  DSCP  5 to priority  0 drop-precedence  2
  DSCP  6 to priority  0 drop-precedence  3
  DSCP  7 to priority  0 drop-precedence  3
  DSCP  8 to priority  7 drop-precedence  0
  DSCP  9 to priority  1 drop-precedence  0
  DSCP 10 to priority  6 drop-precedence  1
  DSCP 11 to priority  1 drop-precedence  1
  DSCP 12 to priority  1 drop-precedence  2
  DSCP 13 to priority  1 drop-precedence  2
  DSCP 14 to priority  1 drop-precedence  3
  DSCP 15 to priority  1 drop-precedence  3
  DSCP 16 to priority  2 drop-precedence  0
  DSCP 17 to priority  2 drop-precedence  0
  DSCP 18 to priority  2 drop-precedence  1
  DSCP 19 to priority  2 drop-precedence  1
  DSCP 20 to priority  2 drop-precedence  2
  DSCP 21 to priority  7 drop-precedence  2
  DSCP 22 to priority  2 drop-precedence  3
  DSCP 23 to priority  2 drop-precedence  3
  DSCP 24 to priority  3 drop-precedence  0
  DSCP 25 to priority  3 drop-precedence  0
  DSCP 26 to priority  3 drop-precedence  1
  DSCP 27 to priority  3 drop-precedence  1
  DSCP 28 to priority  3 drop-precedence  2
  DSCP 29 to priority  3 drop-precedence  2
  DSCP 30 to priority  2 drop-precedence  1
  DSCP 31 to priority  3 drop-precedence  3
   ....
```

Syntax: show qos-map {**dscp** | **exp** | **pcp**} **decode-map** {*map-name* | **all-zero-map** | **default-map**}

The **dscp** option is used to display an Ingress DSCP Policy Map configuration.

The **exp** option is used to display an Ingress EXP Policy Map configuration.

The **pcp** option is used to display an Ingress PCP Policy Map configuration.

The *map-name* variable is the name of the Ingress Policy Map whose configuration you want to display.

The **all-zero-map** option is used to display the specified Ingress Policy Map's all-zero-map configuration.

The **default-map** option is used to display the specified Ingress Policy Map's default configuration.

Displaying QoS Egress Encode Policy Map configurations

To display QoS Egress Encode Policy Map configuration information, enter the following command at any level of the CLI.

```
device(config)# show qos-map dscp encode-map test2
DSCP encode map test2
  Priority  0 drop-precedence  0 to DSCP  0
  Priority  0 drop-precedence  1 to DSCP  2
  Priority  0 drop-precedence  2 to DSCP  4
  Priority  0 drop-precedence  3 to DSCP  6
  Priority  1 drop-precedence  0 to DSCP 44
  Priority  1 drop-precedence  1 to DSCP 44
  Priority  1 drop-precedence  2 to DSCP 44
  Priority  1 drop-precedence  3 to DSCP 44
  Priority  2 drop-precedence  0 to DSCP 20
  Priority  2 drop-precedence  1 to DSCP 25
  Priority  2 drop-precedence  2 to DSCP 20
  Priority  2 drop-precedence  3 to DSCP 20
  Priority  3 drop-precedence  0 to DSCP 55
  Priority  3 drop-precedence  1 to DSCP 55
  Priority  3 drop-precedence  2 to DSCP 55
  Priority  3 drop-precedence  3 to DSCP 55
  Priority  4 drop-precedence  0 to DSCP 32
  Priority  4 drop-precedence  1 to DSCP 34
  Priority  4 drop-precedence  2 to DSCP 36
  Priority  4 drop-precedence  3 to DSCP 38
  Priority  5 drop-precedence  0 to DSCP 54
  Priority  5 drop-precedence  1 to DSCP 54
  Priority  5 drop-precedence  2 to DSCP 54
  Priority  5 drop-precedence  3 to DSCP 54
  Priority  6 drop-precedence  0 to DSCP 48
  Priority  6 drop-precedence  1 to DSCP 50
  Priority  6 drop-precedence  2 to DSCP 52
  Priority  6 drop-precedence  3 to DSCP 54
  Priority  7 drop-precedence  0 to DSCP 27
  Priority  7 drop-precedence  1 to DSCP 27
  Priority  7 drop-precedence  2 to DSCP 27
  Priority  7 drop-precedence  3 to DSCP 27
```

Syntax: show qos-map {**dscp** | **exp** | **pcp**} **encode-map** {*map-name* | **all-zero-map** | **default-map**}

The **dscp** option is used to display an Egress DSCP Policy Map configuration.

The **exp** option is used to display an Egress EXP Policy Map configuration.

The **pcp** option is used to display an Egress PCP Policy Map configuration.

The *map-name* variable is the name of the Egress Policy Map whose configuration you want to display.

The **all-zero-map** option is used to display the specified Egress Policy Map's all-zero-map configuration.

The **default-map** option is used to display the specified Egress Policy Map's default configuration.

Displaying QoS Binding configurations

To display QoS Binding configuration information, enter the following command at any level of the CLI.

```
device(config)# show qos-map binding global
 qos pcp decode-policy pcp-t2
 qos exp decode-policy exp-t1
 qos dscp decode-policy dscp-t3
 qos dscp encode-policy dscp-d3
```

Syntax: show qos-map binding {**global** | *slot/port*}

The **global** option is used to display all QoS Policy Map bindings configured on the device.

The *slot/port* variable is used to display all QoS Policy Map bindings configured on the device.

Displaying QoS packet and byte counters

You can enable and display the collection of statistics for Ingress and Egress packet priorities as described in the following sections:

- Enabling QoS Packet and Byte Counters
- Displaying QoS Packet and Byte Counters
- Clearing QoS Packet and Byte Counters

Enabling QoS packet and byte counters

You can enable the collection of statistics for Ingress and Egress packet priorities using the **enable-qos-statistics** command as shown in the following.

```
device(config)# enable-qos-statistics
device#
```

Syntax: [no] enable-qos-statistics

The default for this command is disabled.

Using the **no** option returns a previous enabled configuration to the default disabled state.

Displaying QoS packet and byte counters

You can enable the collection of statistics for Ingress and Egress packet priorities using the **enable-qos-statistics** command. Once the collection of statistics is enabled, the **show np qos statistics** command can be used to display a count of the packet priorities of Ingress and Egress packets as shown in the following.

```
device# show np qos statistics eth 1/1
Port 1/1
  Ingress counters:
    COS 0: packets 0                     bytes 0
    COS 1: packets 0                     bytes 0
    COS 2: packets 0                     bytes 0
    COS 3: packets 0                     bytes 0
    COS 4: packets 0                     bytes 0
    COS 5: packets 0                     bytes 0
    COS 6: packets 0                     bytes 0
    COS 7: packets 1122084909            bytes 134650189080
  Egress counters:
    COS 0: packets 0                     bytes 0
    COS 1: packets 0                     bytes 0
    COS 2: packets 0                     bytes 0
    COS 3: packets 0                     bytes 0
    COS 4: packets 4056756685            bytes 486810801752
    COS 5: packets 0                     bytes 0
    COS 6: packets 0                     bytes 0
    COS 7: packets 453                   bytes 49490
```

Syntax: show np qos statistics {**ethernet** *slot/port* | **slot** *slot-number*}

The **ethernet** option is used to display all QoS counters for the ethernet interface specified by the *slot/port* variable.

The **slot** option is used to display all QoS counters for the interface module whose location is specified by the *slot-number* variable.

TABLE 28 QoS counter information

This field...	Displays...
Ingress Counters	Statistics displayed below this heading are for packets arriving on the Ingress port (or ports) before any overrides or merging of packet priorities have been performed.
COS *num*: packets	The number of packets that have arrived on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the *num* variable.
COS *num*: bytes	The number of bytes contained in the packets that have arrived on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the *num* variable.
Egress Counters	Statistics displayed below this heading are for packets leaving the device on the Egress port (or ports) accounting for all priority modifications that have been performed on them. These statistics accurately reflect the values for packets that are forwarded out of the device.
COS *num*: packets	The number of packets leaving the device on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the *num* variable.
COS *num*: bytes	The number of bytes contained in the packets leaving the device on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the *num* variable.

Clearing the QoS packet and byte counters

You can clear the QoS counters whose display is generated using the **show np qos statistics** command as shown in the following.

```
device(config)#clear np qos statistics ethernet 2/5
```

Syntax: clear np qos statistics ethernet *slot/port*

The **ethernet** option is used to clear all QoS counters for the ethernet interface specified by the *slot/port* variable.

Weighted Random Early Discard (WRED)

On the Brocade device, queues are provided to buffer traffic levels that exceed the bandwidth of individual ports. For each output port, a set of eight priority queues is allocated on each inbound traffic manager. When traffic exceeds the bandwidth of a port, packets are dropped randomly as long as the congestion persists. Under these conditions, traffic of greater priority can be dropped instead of traffic with a lesser priority.

Instead of being subject to this random process, you can configure a Brocade device to monitor traffic congestion and drop packets according to a WRED (Weighted Random Early Discard) algorithm. This algorithm enables the system to detect the onset of congestion and take corrective action. In practice, WRED causes a device to start dropping packets as traffic in the device starts to back up. WRED provides various control points that can be configured to change a system's reaction to congestion. The following variables are used when calculating whether to drop or forward packets:

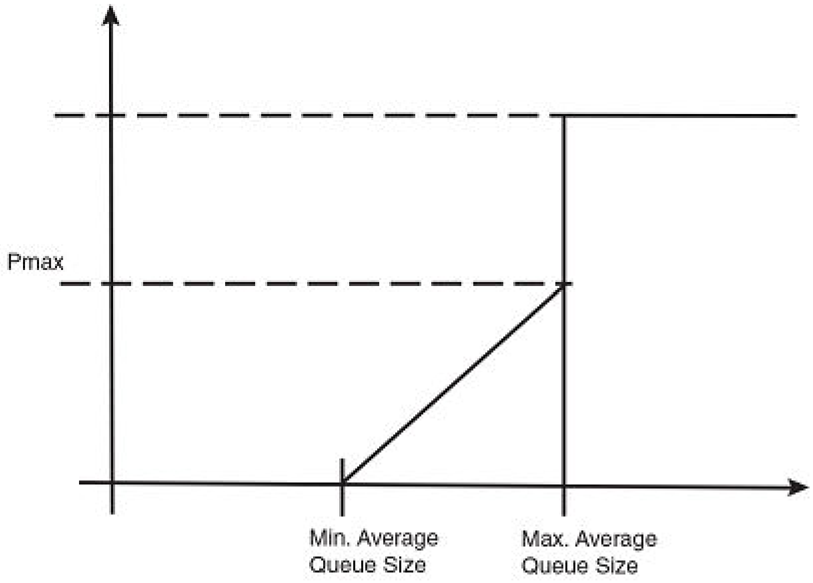
- **Statistical Average-Q-Size** - The statistical average size of the queue calculated over time on the device.
- **Current-Q-Size** - The current size of the queue as calculated on the device.

- **Wq** - This variable specifies the weights that should be given to the current queue size and the statistical average-q-size when calculating the size for WRED calculations.
- **Max-Instantaneous-Q-Size** - The maximum size up to which a queue is allowed to grow. Packets that cause the queue to grow beyond this point are unconditionally dropped. This variable is user configured.
- **Min-Average-Q-Size** - The average queue size below which all packets are accepted. This variable is user configured.
- **Max-Average-Q-Size** - The average queue size above which all packets are dropped. This variable is user configured.
- **Pmax** - The maximum drop probability when queue-size is at Max-Average-Q-Size. This variable is user configured.
- **Pkt-Size-Max** - The packet size to which the current packet's size is compared as shown in the algorithm below. This variable is user configured.

How the WRED algorithm operates

The WRED operation graph below describes the interaction of the previously described variables in the operation of WRED. When a packet arrives at a device, the average queue size (**avg-q-size**) is calculated as described below (note that this is not the statistical average queue size). If **avg-q-size** as calculated, is below the configured Min. Average Queue Size, then the packet is accepted. If the average queue size is above the Max. configured Average Queue Size threshold, the packet is dropped. If the instantaneous queue size exceeds the value configured for the Max-Instantaneous-Q-Size, the packet is dropped. If the Average Queue size falls between the Min. Average Queue Size and the Max. Average Queue Size, packets are dropped according to the calculated probability described below.

FIGURE 3 WRED operation graph



Calculating avg-q-size

The algorithm first calculates the **avg-q-size** through the following equation.

avg-q-size = ((1 - Wq) * Statistical Average-Q-Size) + (Wq * Current-Q-Size)

The user-configured **Wq** value is instrumental to the calculation and can be:

- equal to the statistical average queue size (**Wq** == 0), or
- equal to the current queue size (**Wq** == 1) or
- be between 0 and 1 (0 < **Wq** < 1).

Lower Wq values cause the **avg-q-size** to lean towards the statistical average queue size, reducing WRED's sensitivity to the current state of the queue and thus reducing WRED's effectiveness. On the other hand, higher **Wq** values cause the **avg-q-size** to lean towards the instantaneous queue size, which exposes WRED to any change in the instantaneous queue size and thus may cause WRED to overreact in cases of bursts. Thus, the value of **Wq** should be carefully chosen according to the application at hand.

Calculating packets that are dropped

The **Pdrop** value, as calculated in the following equation, is the probability that a packet will be dropped in a congested device.

	pkt-size		**(avg-q-size - min-avg-q size)**
Pdrop =	**-----------------**	*** Pmax ***	**---**
	pkt-size-max		**(max-avg-q-size - min-avg-q size)**

Applying the WRED algorithm to device traffic

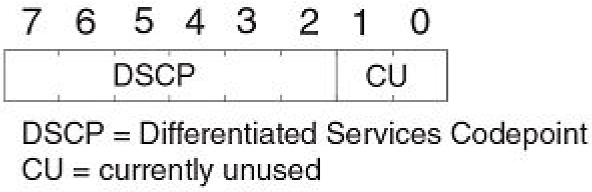
Packets are assigned to an Ingress queue type based on their individual destination port and one of the 8 (0 - 7) internal priorities. Each of these priorities is assigned a queue type from 0 - 7 according to the internal priority it belongs to.

TABLE 29 Internal priority to queue type mapping

Internal priority	0	1	2	3	4	5	6	7
Queue type	0	1	2	3	4	5	6	7

The WRED algorithm is applied to traffic on these individual queues based upon parameters configured for its assigned queue type. When traffic arrives at a queue, it is passed or dropped as determined by the WRED algorithm. Packets in an individual queue are further differentiated by one of four drop precedence values which are determined by the value of bits 3:2 of the TOS or DSCP bits in the IPv4 or IPv6 packet header.

FIGURE 4 TOS or DSCP bits in packet header



The user configurable values applied per queue type and per drop precedence value are:

- Maximum Drop Probability
- Minimum and Maximum Average Queue Size
- Maximum Packet Size

Configuring packet drop priority using WRED

For a description of WRED, refer to Weighted Random Early Discard (WRED) on page 152. This section describes how to configure the parameters described in that section to enable the use of WRED on a Brocade device. In addition, there is a default configuration that can be enabled that sets the parameters to the values shown in the WRED default settings table in Restoring default WRED parameters on page 159. If you use the default configuration, you do not need to set the parameters individually.

Enabling WRED

WRED must be enabled for the queue type of any forwarding queue that you want it to operate on. To enable WRED for the forwarding queues with a queue type of 3, enter the following command.

```
device(config)#qos queue-type 3 wred enable
```

Syntax: [no] qos queue-type *queue-number* **wred enable**

The *queue-type* variable is the number of the forwarding queue that you want to enable WRED for. There are eight forwarding queues on Brocade devices. They are numbered 0 to 7. Default values are as described in the WRED default settings table in Restoring default WRED parameters on page 159. You can optionally adjust any of the pre-configured parameters described there.

Setting the averaging-weight (Wq) parameter

The Wq parameter is configured as the **averaging-weight** parameter. In this implementation, you can set one of 13 (1 - 13) possible values. These values represent a Wq value as described in the following table.

TABLE 30 Possible Wq values

Averaging weight setting	Wq value as a percentage
1	50%
2	25%
3	12.5%
4	6.2%
5	3.12%
6	1.56%
7	0.78%

TABLE 30 Possible Wq values (Continued)

Averaging weight setting	Wq value as a percentage
8	0.4%
9	0.2%
10	0.09%
11	0.05%
12	0.02%
13	0.01%

To set the wq parameter for queues with a queue type of 1 to 25%, use the following command.

```
device(config)#qos queue-type 1 wred averaging-weight 2
```

This gives the current queue size a weight of 25% over the statistical average queue size.

Syntax: [no] qos queue-type *queue-type* **wred averaging-weight** *avg-weight-value*

The *queue-type* variable is the number of the forwarding queue type that you want to configure the **averaging-weight** (Wq) parameter for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *avg-weight-value* variable is the weight-ratio between instantaneous and average queue sizes. Valid values are 1 through 13. The default value is 9 which maps to a Wq value of .2%.

Configuring the maximum instantaneous queue size

You can set the maximum size to which a queue is allowed to grow. Packets that cause the queue to grow beyond this setting are unconditionally dropped. To set the maximum instantaneous queue size for queues with a queue type of 1 to 32000 KBytes, use the following command.

```
device(config)#qos queue-type 1 max-queue-size 32
```

Syntax: [no] qos queue-type *queue-number* **max-queue-size** *max-queue*

The *queue-type* variable is the number of the forwarding queue type that you want to configure the **instantaneous-queue-size** parameter for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *max-queue* variable is the maximum size to which a queue is allowed to grow. It is defined in Kbytes. The default values are shown in the WRED default settings table in Restoring default WRED parameters on page 159.

Configuring the drop precedence parameters

The DSCP or TOS bits in packets are used to prioritize packet delivery for specified queue types. These values are from 0 to 4. Packets with a DSCP or TOS value of 0 are least likely to be dropped and packets with a DSCP or TOS of 3 are most likely to be dropped.

NOTE
In addition to bits in the DSCP, the DP option can use other fields (in the PCP header or the EXP bit header) to control WRED in the priority queues.

In addition, the maximum drop probability, the minimum and maximum average queue size, and the maximum packet size can be configured to apply selectively to packets with a specified queue type and DSCP or TOS value. The following sections describe how to set the drop precedence parameters **drop-probability-max**, **max-avg-queue-size**, **min-avg-queue-size** and **packet-size-max** for each of the four DSCP or TOS values for each of the four queue types.

NOTE
Packets that do not have the DSCP or TOS value set are assigned a drop precedence equal to the DSCP or TOS level of 0.

Setting the maximum drop probability

To set the maximum drop probability for queue type 1 and drop precedence 0 when the queue size reaches the Max-average-q-size value to 20% use the following command.

```
device(config)#qos queue-type 1 wred drop-precedence 0 drop-probability-max 20%
```

Syntax: [no] qos queue-type *queue-type* **wred drop-precedence** *drop-precedence-value* **drop-probability-max** *p-max*

The *queue-type* variable is the number of the forwarding queue type that you want to configure drop-precedence for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *drop-precedence-value* variable for the drop-precedence parameter is the TOS or DSCP value in the IPv4 or IPv6 packet header. It determines drop precedence on a scale from 0 - 3. Packets than contain a DSCP value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped. The default value is 0 .

The *p-max* variable defines the maximum drop probability when the queue size is at the value configured for **max-avg-q-size**. This value is expressed as a percentage. Use the % sign after you type the drop-probability-max value. The default values are shown in the WRED default settings table in Restoring default WRED parameters on page 159.

Setting the minimum and maximum average queue size

When setting the minimum and maximum average queue size, consider the following

- If a user enters a min-avg-queue-size that is equal to what is currently configured for the max-avg-queue-size, then the min-avg-queue-size is decremented by 64. The min-avg-queue-size is decremented by 64 because the value must be different from the max-avg-queue-size that is currently configured. The following example is a warning message that is displayed on the console.

```
Warning: The min-avg-queue-size is decreased to(min-avq-queue-size-64)as min and max
should be different to be effective.
```

- If a user enters a max-avg-queue-size that is equal to what is currently configured for the min-avg-queue-size, then the max-avg-queue-size is incremented by 64. The max-avg-queue-size is incremented by 64 because the value must be different from the min-avg-queue-size that is currently configured. The following example is a warning message that is displayed on the console.

```
Warning: The max-avg-queue-size is increased to(max-avq-queue-size+64)as min and max
should be different to be effective.
```

- If a user enters a min-avg-queue-size equal to the max-avg-queue-size, then the max-avg-queue-size is incremented by 64. The max-avg-queue-size is incremented by 64 because the value must be different from the min-avg-queue-size The following example is a warning message that is displayed on the console.

```
Warning: The max-avg-queue-size is increased to(max-avq-queue-size+64)as min and max
should be different to be effective.
```

However if a user enters a max-avg-queue-size and min-avg-queue-size equal to 32768, then the min-avg-queue-size is decremented.

To set the maximum average queue size for queue type 1 and drop precedence 0 to the maximum size of 32768 Kbytes, use the following command.

```
device(config)#qos queue-type 1 wred drop-precedence 0 max-avg-queue-size 32768
```

Syntax: no qos queue-type *queue-type* **wred drop-precedence** *drop-precedence-value* **max-avg-queue-size** *max-size*

To set the minimum average queue size to the maximum size of 16 Kbytes, use the following command.

```
device(config)#qos queue-type 1 wred drop-precedence 0 min-avg-queue-size 16
```

Syntax: [no] qos queue-type *queue-type* **wred drop-precedence** *drop-precedence-value* **min-avg-queue-size** *min-size*

The *queue-type* variable is the number of the forwarding queue type that you want to configure drop-precedence for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *drop-precedence-value* variable for the drop-precedence parameter is the TOS or DSCP value in the IPv4 or IPv6 packet header. It determines drop precedence on a scale from 0 - 3 . Packets than contain a DSCP value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped. The default value is 0.

The *min-size* variable is the average queue size below which all packets are accepted. Possible values are 1 - 32768 KBytes. It must be set in multiples of 64K. The default values are shown in the WRED default settings table in Restoring default WRED parameters on page 159.

The *max-size* variable is the average queue size above which all packets are dropped. (1 - 32768) (KBytes) in multiples of 64K. The default values are shown in the WRED default settings table in Restoring default WRED parameters on page 159.

Setting the maximum packet size

To set the maximum packet size to 16 bytes for queue type 1 and drop precedence 0, use the following command.

```
device(config)#qos queue-type 1 wred drop-precedence 0 packet-size-max 16
```

Syntax: [no] qos queue-type *queue-type* **wred drop-precedence** *drop-precedence-value* **packet-size-max** *pkt-size*

The *queue-type* variable is the number of the forwarding queue type that you want to configure drop-precedence for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *drop-precedence-value* variable for the drop-precedence parameter is the TOS or DSCP value in the IPv4 or IPv6 packet header. It determines drop precedence on a scale from 0 - 3. Packets than contain a DSCP value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped. The default value is 0.

The *pkt-size* variable is the pkt-size-max variable used in the equation described in How the WRED algorithm operates on page 153. Permissible values are an even number of bytes between 16 and 32768. The default values are shown for each queue type and drop precedence value in the WRED default settings table in Restoring default WRED parameters on page 159.

Restoring default WRED parameters

The table below describes all of the default values for each of the WRED parameters. If you change any of the values from the default values, you can restore the defaults per queue type. To reset the queue type 1 with default values for the WRED parameters, use the following command.

```
device(config)#qos queue-type 1 wred default-params
```

Syntax: [no] qos queue-type *queue-number* **default-params**

The *queue-number* variable is the number of the forwarding queue that you want to configure drop-precedence for. There are eight forwarding queues on Brocade devices. They are numbered 0 to 7

TABLE 31 WRED default settings

Queue type	Drop precedence	Minimum average queue size (KByte)	Maximum average queue size (KByte)	Maximum packet size (Byte)	Maximum drop probability	Maximum instantaneous queue size (Kbyte)	Average weight
0	0	320	1024	16384	2%	1024	6.25%
	1	256	1024	16384	4%		
	2	256	1024	16384	9%		
	3	192	1024	16384	10%		
1	0	320	1024	16384	2%	1024	6.25%
	1	256	1024	16384	4%		
	2	256	1024	16384	9%		
	3	192	1024	16384	9%		
2	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		

TABLE 31 WRED default settings (Continued)

Queue type	Drop precedence	Minimum average queue size (KByte)	Maximum average queue size (KByte)	Maximum packet size (Byte)	Maximum drop probability	Maximum instantaneous queue size (Kbyte)	Average weight
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
3	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
4	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
5	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
6	0	1024	1088	16384	0%	1024	6.25%
	1	448	832	16384	2%		
	2	384	832	16384	5%		
	3	320	832	16384	6%		
7	0	1024	1088	16384	0%	1024	6.25%
	1	448	832	16384	2%		
	2	384	832	16384	5%		
	3	320	832	16384	6%		

NOTE

- If you enter the *min-avg-queue-size* equal to what is already configured as the *max-avg-queue-size* , then the *min-avg-queue-size* will be decremented by 64 to make it different from the *max-avg-queue-size* , the following warning is displayed: "Warning - *min-avg-queue-size* is decreased to (*min-avg-queue-size* - 64) as min and max should be different to be effective."
- If you enter the *max-avg-queue-size* equal to what is already configured as the *min-avg-queue-size*, then the *max-avg-queue-size* will be incremented by 64 to make it different from the *min-avg-queue-size* , the following warning is displayed:"Warning - *max-avg-queue-size* is increased to (*max-avg-queue-size* + 64) as the min & max should be different to be effective."
- If you enter the min-average-queue-size equal to the *max-avg-queue-size* , the *max-avg-queue-size* will be incremented by 64 to make it different from *min-avg-queue-size*, the following warning is displayed: "Warning - *max-avg-queue-size* increased to (*max-avg-queue-size* + 64) as min & max should be different to be effective." Unless you enter the *max-avg-queue-size* and *min-avg-queue-size* equal to 32768, the *min-avg-queue-size* will be decremented.

Displaying the WRED configuration

To view a WRED configuration, use the following command.

```
device# show qos wred
QType Enable AverWt    MaxQSz DropPrec MinAvgQSz MaxAvgQSz MaxDropProb MaxPktSz
0     Yes    9(0.19%)  16384  0             5696     16384          2%      16384
                              1             4864     16384          4%      16384
                              2             4096     16384          9%      16384
                              3             3264     16384         10%      16384
1     No
2     No
3     Yes    9(0.19%)  16384  0             6528     16384          2%      16384
                              1             5696     16384          4%      16384
                              2             4864     16384          9%      16384
                              3             4096     16384          9%      16384
4     No
5     No
6     No
7     No
```

Scheduling traffic for forwarding

If the traffic being processed by a Brocade device is within the capacity of the device, all traffic is forwarded as received. Once we reach the point where the device is bandwidth constrained, it becomes subject to drop priority if configured as described in Configuring packet drop priority using WRED on page 155 or traffic scheduling as described in this section.

The Brocade devices classify packets into one of eight internal priorities. Traffic scheduling allows you to selectively forward traffic according to the forwarding queue that is mapped to according to one of the following schemes:

- **Strict priority-based scheduling** - This scheme guarantees that higher-priority traffic is always serviced before lower priority traffic. The disadvantage of strict priority-based scheduling is that lower-priority traffic can be starved of any access.
- **WFQ weight-based traffic scheduling** - With WFQ destination-based scheduling enabled, some weight-based bandwidth is allocated to all queues. With this scheme, the configured weight

distribution is guaranteed across all traffic leaving an egress port and an input port is guaranteed allocation in relationship to the configured weight distribution.

- **Mixed strict priority and weight-based scheduling** - This scheme provides a mixture of strict priority for the three highest priority queues and WFQ for the remaining priority queues.

Configuring traffic scheduling

Traffic scheduling can be configured on a per-port basis. It affects the outgoing traffic on the configured port when bandwidth congestion occurs on that port. The following sections describe how to configure each of the traffic scheduling schemes.

Configuring strict priority-based traffic scheduling

This option is the default traffic scheduling method if traffic scheduling is not configured on a port

To configure strict priority-based scheduling use a command such as the following.

```
device(config)# interface ethernet 1/1
device(config-if-e1000-1/1)# qos scheduler strict
```

Syntax: qos scheduler strict

This is the default when traffic scheduling is not configured.

Calculating the values for WFQ Weight-based traffic scheduling

Weighted Fair Queueing (WFQ) scheduling is configured to be a percentage of available bandwidth using the following formula.

	q (x)
Weight of q (x) =	**--**
	q0 + q1 + q2 +q3+ q4 + q5 +q6 +q7

Where

q (x) = The value of the queue that you want to determine the weight for. It can be the value of any queue (0 - 7).

q0 - q7 = the assigned values of the eight queues.

Weight of q (x) = the calculated weight as a percentage of the port's total bandwidth.

For example if you assign the following values to queues 0 to 7:

- Queue 0 =10, Queue 1 = 15, Queue 2 = 20, Queue 3 = 25, Queue 4 = 30, Queue 5 = 35, Queue 6 = 40, and Queue 7 = 45,

To determine the weight of **q3** , use the following formula.

	25
Weight of q3 =	**--**

10 + 15 + 20 + 25 + 30 + 35 + 40 + 45

The weight of q3 is 11.4%. Consequently, q3 will get 11.4% of the port's total bandwidth.

The values of the remaining queues are calculated to be the following: q7 = 20.5%, q6 = 18.2%, q5 = 15.9%, q4 = 13.6%, q3 = 11.4%, q2 = 9.1%, q1 = 6.8%, and q0 = 4.5%

Configuring WFQ weight-based traffic scheduling

To configure WFQ weight-based scheduling use a command such as the following. The actual weights applied to the hardware are displayed.

```
device(config)# interface ethernet 1/1
device(config-if-e1000-1/1)# qos scheduler weighted 100 20 39 50 60 70 80 10

Weights applied are 10 2 3 5 6 7 8 1
```

Syntax: qos scheduler weighted *queue7-weight queue6-weight queue5-weight queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*

The *queue7-weight* variable defines the relative value for queue7 in calculating queue7's allocated bandwidth.

The *queue6-weight* variable defines the relative value for queue6 in calculating queue6's allocated bandwidth.

The *queue5-weight* variable defines the relative value for queue5 in calculating queue5's allocated bandwidth.

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The *queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth

The acceptable range for *queuex-weight* variables is 1-128.

Refer to Calculating the values for WFQ Weight-based traffic scheduling on page 162 for information on assigning queue0-weight to queue7-weight values.

Configuring mixed strict priority and weight-based scheduling

When configuring the mixed strict priority and weight-based scheduling option, queues 5 - 7 are allocated to strict priority-based scheduling and queues 0 - 4 are allocated to weight-based scheduling.

To configure mixed priority and weight-based scheduling use a command such as the following. The actual weights applied are displayed.

```
device(config)# interface ethernet 1/1
device(config-if-e1000-1/1)# qos scheduler mixed 100 20 30 4 5

Weights applied are 26 5 7 1 1
```

Syntax: qos scheduler mixed *queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The *queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth

The acceptable range for *queuex-weight* variables is 1-128.

Refer to Calculating the values for WFQ Weight-based traffic scheduling on page 162 for information on assigning queue0-weight to queue4-weight values.

Configuring egress unicast and multicast traffic scheduling

You can schedule the egress unicast and multicast traffic based on the allocated bandwidth. To configure WFQ weight-based scheduling for an interface, enter the following commands.

```
device(config)# interface ethernet 1/1
device(config-if-e1000-1/1)# qos egress-weight unicast 1 multicast 3
```

Syntax: [no] qos egress-weight unicast *unicast-weight-val* **multicast** *multicast-weight-val*

The **unicast** *unicast-weight-val* variable specifies the allocated bandwidth for the egress unicast traffic. The value ranges from 1 through 255.

The **multicast** *multicast-weight-val* variable specifies the allocated bandwidth for the egress multicast traffic. The value ranges from 1 through 255.

NOTE
Egress scheduling will not work properly if ingress multicast shaper is configured to full rate and high multicast traffic is present. To provide better functionality, reduce the rate of ingress multicast shaper.

Displaying QoS scheduler configuration

Review the QoS scheduler configuration using the **show qos scheduler** command.

The following is an example of the **show qos scheduler** command.

```
device(config-if-e100000-2/1)#show qos scheduler
CW: Configured Weight   AW: Applied Weight (Valid for Weighted and Mixed schemes
only)
Port   | Scheme   Type   Pri7    Pri6    Pri5    Pri4    Pri3    Pri2    Pri1
Pri0
-------+----------------+-------+-------+-------+-------+-------+-------+-------
+-------
   1/1 | strict
   1/2 | strict
   1/3 | strict
   1/4 | strict
   1/5 | strict
   1/6 | strict
   1/7 | strict
   1/8 | strict
   1/9 | strict
  1/10 | strict
  1/11 | strict
  1/12 | strict
  1/13 | strict
  1/14 | strict
  1/15 | strict
  1/16 | strict
  1/17 | strict
  1/18 | strict
  1/19 | strict
  1/20 | strict
  1/21 | strict
  1/22 | strict
  1/23 | strict
  1/24 | strict
   2/1 | mixed    CW/AW                          100/26   20/5    30/7     4/1
5/1
   2/2 | strict
   3/1 | strict
   3/2 | strict
   3/3 | strict
   3/4 | strict
   3/5 | strict
   3/6 | strict
   3/7 | strict
   3/8 | strict
   3/9 | strict
  3/10 | strict
  3/11 | strict
  3/12 | strict
  3/13 | strict
  3/14 | strict
  3/15 | strict
  3/16 | strict
  3/17 | strict
  3/18 | strict
  3/19 | strict
  3/20 | strict
   4/1 | weighted CW/AW  100/125   1/1    25/25    1/1    50/50    1/1     1/1
75/83
   4/2 | strict
   4/3 | strict
   4/4 | strict
   5/1 | strict
   5/2 | strict
   5/3 | strict
   5/4 | strict
   6/1 | strict
   6/2 | strict
   6/3 | strict
   6/4 | strict
   6/5 | strict
   6/6 | strict
   6/7 | strict
   6/8 | strict
   7/1 | strict
   7/2 | strict
device(config-if-e100000-2/1)#
```

The following is an example of the **show qos scheduler ethernet** command.

```
device(config-if-e100000-2/1)#show qos scheduler ethernet 2/1
CW: Configured Weight   AW: Applied Weight (Valid for Weighted and Mixed schemes only)
Port   | Scheme   Type   Pri7    Pri6    Pri5    Pri4    Pri3    Pri2    Pri1
Pri0
-------+----------------+-------+-------+-------+-------+-------+-------+-------
+-------
   2/1 | mixed    CW/AW                          100/26   20/5    30/7     4/1
5/1
device(config-if-e100000-2/1)#
```

Egress port and priority-based rate shaping

Rate shaping is a mechanism to smooth out the variations in traffic above a certain rate. The primary difference between rate shaping and rate limiting is that in rate limiting, traffic exceeding a certain threshold is dropped. In rate shaping, the traffic that exceeds a threshold is buffered so that the output from the buffer follows a more uniform pattern. Rate shaping is useful when burstiness in the source stream needs to be smoothed out and a more uniform traffic flow is expected at the destination.

NOTE
Because excess traffic is buffered, rate shaping must be used with caution. In general, it is not advisable to rate shape delay-sensitive traffic.

Brocade devices support egress rate shaping. Egress rate shaping is supported per port or for each priority queue on a specified port.

Configuring port-based rate shaping

When setting rate shaping for a port, you can limit the amount of bandwidth available on a port within the limits of the port's rated capacity. Within that capacity, you can set the bandwidth at increments within the ranges described in the following table.

TABLE 32 Port-based rate shaping interval table

Range	Increment supported within the range
0 - 10Mb	8,333
10Mb - < 100Mb	20,833
100 Mb - < 1Gb	208,333
1Gb - 10Gb	2,083,333

NOTE
The egress rate shaping burst size for a port-based shaper is 10,000 bytes.

These limits provide a minimum and maximum rate to which the port can be set. They also provide the increments at which the port capacity can be set. In operation, you can set any number between the minimum and maximum values. The device will automatically round-up the value to the next higher increment.

For example, if you set the rate of a 10G port to 2,000,000,000, the actual rate would be 2,002,083,173. This is because it is the next highest increment above 2,000,000,000.

To set a 10 Gbps port to the incremental port capacity over 2 Gbps, use the following command.

```
device(config)# interface ethernet 2/2
device(config-if-e10000-2/2)# qos shaper 2000000000
```

Syntax: no qos shaper *rate*

The *rate* variable sets the rate for the port within the limits available as described in the table above. The rate is set in bps.

NOTE
When the rate shaping of a port is enabled, you can expect a deviation of plus or minus 3% actual port rate from the configured port rate. The deviation may vary between different types of TM devices within the ranges mentioned in the table. The plus or minus 3% deviation applies to unicast and multicast, default and user-configured shaper rates. In addition, if you configure port shaper on 24x10, 2x100, and 8x10 modules for sizes of 8 to 400 MB, the result will not be deterministic and priorities may not be guaranteed for the bandwidth configured. A shaper configuration less than 8 MB is not supported.

Configuring port and priority-based rate shaping

When setting rate shaping for a priority queue, you can limit the amount of bandwidth available for a specified priority within the limits of the capacity of the port that the priority is configured on. You can set the limit for the priority to any value from 1 to the port's maximum rating and the device will automatically round up the value to the next increment supported. This will be a slightly higher value than what you specify with the command. For example, if you set the rate for priority 2 on a 10G port to 2,000,000,000, the actual rate would be slightly higher.

NOTE
The egress rate shaping burst size for a port and priority-based shaper is 3072 bytes.

To set the capacity for priority 2 traffic on a 10 Gbps port to the incremental capacity over 2 Gbps, use the following command.

```
device(config)# interface ethernet 2/2
device(config-if-e10000-2/2)# qos shaper priority 2 2000000000
```

Syntax: [no] qos shaper priority *priority-level rate*

The *priority-level* variable specifies the priority that you want to set rate shaping for on the port being configured.

The *rate* variable sets the rate you want to set for the priority. The rate is set in bps.

Multicast queue size, flow control, rate shaping, and egress buffer threshold

There are four internal priorities for multicast or broadcast traffic. These four priorities are mapped from the device's eight internal priorities.

TABLE 33 Mapping between multicast or broadcast and internal forwarding priorities

Internal Forwarding Priority	0,1	2,3	4,5	6,7
Multicast Internal Priority	0	1	2	3

The internal forwarding priority of a multicast or broadcast packet is determined from the packet's IEEE 802.1p priority, incoming port priority or IP ToS or DSCP, as described in Default QoS mappings on page 112. Four multicast queue types (0 to 3) are used for multicast internal priorities 0 to 3 respectively.

NOTE
Instead of ACL priority, use VLAN or port priority to prioritize multicast traffic.

Configuring multicast queue size

The following example configures a 2 MB queue size for queue 0.

```
device(config)# qos multicast-queue-type 0 max-queue-size 2048
```

Syntax: [no] qos multicast-queue-type *queue-number* **max-queue-size** *queue-size*

The *queue-number* variable specifies the queue for which you want to configure a maximum size. The possible values are 0, 1, 2, and 3.

The *queue-size* variable specifies the size in KB that you want to set as the maximum value for the specified multicast queue. The possible values are 1 through 32768 KB. The default queue size is 1 Mb.

This command is applied per device and takes effect on all Traffic Managers within the configured device.

NOTE
The **qos multicast-queue-type** command is not applicable on the Brocade CES and Brocade CER platforms.

Configuring multicast flow control

Flow controls are available from egress to ingress, and from fabric to ingress. At the egress of each Traffic Manager, there are predetermined thresholds for consumed resources and available resources, and separate thresholds for guaranteed multicast or broadcast traffic and best-effort multicast or broadcast traffic. When a threshold is crossed, flow control can be triggered and multicast or broadcast traffic of the corresponding class is stopped at ingress until resources are below the threshold again. Flow control is disabled by default and can be enabled on an interface by using the following command.

```
device(config)# interface ethernet 2/2
device(config-if-e10000-2/2)# qos multicast flow-control
```

Syntax: no qos multicast flow-control

This command changes the flow control setting on the Traffic Manager where the interface resides. Using this command, Multicast Congestion Indication (MCI) flow control can be enabled (or disabled) which will disable (or enable) ERP flow control on that Traffic Manager.

NOTE
When the system boots up, the ERP flow control (applicable to ingress-replicated traffic) will be enabled for multicast traffic by default.

NOTE
The **qos multicast flow-control** command should be used with caution because MCI affects all the Traffic Managers in the chassis.

Use the no form of the command to disable multicast flow control.

Syntax: no qos multicast flow-control [**threshold** *grntd* % *best-effort* %]

This **qos multicast flow-control** contains an option to set the thresholds for guaranteed and best-effort multicast traffic in terms of percentage of egress resources used. These threshold values are the same for MCI and ERP flow control.

The threshold determines how fast the system responds to multicast oversubscription on the egress Traffic Manager.

Use the no form of the command to disable multicast flow control.

NOTE
The **qos multicast flow-control** command is not applicable on the Brocade CES and Brocade CER platforms.

Configuring multicast rate shaping

You can specify either guaranteed or best-effort multicast rate shaping for a port in kilobits per second (Kbps). Multicast rate shaping is configured per-port to the Ingress port.

The following example changes the best-effort multicast traffic rate to 10 Mbps.

```
device(config)# interface ethernet 2/2
device(config-if-e10000-2/2)# qos multicast shaper best-effort rate 10000
```

Syntax: [no] qos multicast shaper {**guaranteed** | **best-effort**} **rate** *bandwidth*

The **guaranteed** option specifies that the multicast or broadcast shaper applies only to internal multicast priority 3 (the highest multicast priority) traffic.

The **best-effort** option specifies that the multicast or broadcast shaper applies to internal multicast priority 0, 1, and 2 traffic only.

The *bandwidth* variable specifies the maximum bandwidth in Kbps for guaranteed or best-effort multicast traffic scheduled by the Traffic Manager across the switch fabric.

NOTE
The **qos multicast shaper** command is not applicable on the Brocade CES and Brocade CER platforms.

Configuration considerations for multicast rate shaping

When applied to a port, the **qos multicast shaper** command configuration is applied to all ports on the interface module that use the same Traffic Manager as the configured port. This is unlike the behavior of rate shaping applied for unicast traffic. The relationship between ports and Traffic Managers is defined in Displaying traffic manager statistics for an interface module on page 179.

Example 1

In the following example, multicast rate shaping is applied to port 1/18 on a 20-port, 10/100/1000 copper Ethernet interface module (NI-XMR-1Gx20-GC).

```
device(config)# interface ethernet 1/18
device(config-if-e1000-1/18)# qos multicast shaper best-effort rate 10000
```

In this example, the configuration will apply to ingress traffic that arrives on any port of the interface module.

Example 2

In the following example, multicast rate shaping is applied to port 1/1 of a 4-port, 10 GbE Ethernet interface module (NI-XMR-10Gx4).

```
device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# qos multicast shaper best-effort rate 10000
```

In this example, the configuration will apply to ingress traffic that arrives on either port 1/1 or port 1/2 of the interface module.

NOTE
When a **qos multicast shaper** command is configured for a port, the configuration command is placed in the running configuration for all ports that belong to the same Traffic Manager. In Example 1, the **qos multicast shaper best-effort rate 10000** command would appear in the interface configuration section for all ports (1 to 20) on the interface module. In Example 2, the **qos multicast shaper best-effort rate 10000** command would appear in the interface configuration section for ports 1 and 2 on the interface module.

Configuring multicast and unicast egress buffer threshold

You can set the multicast and unicast egress buffer threshold up to 95 percent of the total egress buffer size. This helps in reducing the egress packet drops when there is high multicast traffic. The default configuration egress buffer threshold per port is set to 50 percent of the total egress buffer size.

NOTE
It is recommended to use the **qos multicast egress-max-buffer port** command when a sudden burst is seen in multicast or unicast traffic. This command should only be used at the recommendation of Brocade technical support.

NOTE
The multicast and unicast egress buffer threshold commands are not applicable on the Brocade CES and Brocade CER platforms.

The following example shows how to configure the egress multicast buffer threshold for individual ports so that each port has its dedicated buffer space.

The example uses an 8x10 line card and has four ports per Traffic Manager. The buffer size is divided into four times the total buffer size so that each port has its dedicated buffer space.

```
device #config terminal
device (config)#interface ethernet 1/1
device config-if-e10000-1/1)#qos multicast egress-max-buffer port 24% 23%
```

Syntax: qos multicast egress-max-buffer port { *guaranteed_max_buffer* | *best-effort_max_buffer* }

The *guaranteed_max_buffer* variable specifies the egress buffer size per port for guaranteed traffic flow (multicast port priority 3). Specified as percentage of total buffer size per port, the guaranteed maximum buffer size covers from 1 through 95 percent. Default is 60 percent.

The *best-effort_max_buffer* variable specifies the egress buffer size per port for best effort traffic flow (multicast port priorities 0-2). Specified as a percentage of total buffer size per port, best effort covers 1 through 95 percent. Default is 50 percent.

The following example shows how to set the unicast buffer threshold that a single port on the Traffic Manager can utilize. The example uses a 20x10G line card that has two Traffic Managers. There are 10 ports per Traffic Manager. Each port has a buffer threshold value of 10 percent and 9 percent.

```
device#config terminal
device(config)#interface ethernet 1/1
device(config-if-e10000-1/1)#qos egress-max-buffer port 10% 9%
```

Syntax: qos egress-max-buffer port { *guaranteed_max_buffer* | *best-effort_max_buffer* }

The *guaranteed_max_buffer* variable specifies the egress buffer size per port for guaranteed traffic flow (unicast priorities 6-7). Specified as percentage of total buffer size per port, guaranteed covers 1 through 95 percent. Default is 60 percent.

The *best-effort_max_buffer* variable specifies the egress buffer size per port for best effort traffic flow (unicast priorities 0-5). Specified as percentage of total buffer size per port, best effort covers 1 through 95 percent. Default is 50 percent.

Configuring ingress replication of multicast packets

Ingress Replication resolves the issue arising out of the "unscheduled" nature of multicast traffic, because multicast traffic must wait for credits from the egress Traffic Manager (TM) or Egress Replication Port (ERP).

The following example enables ingress replication of multicast packets corresponding to the VLAN.

```
device# configure terminal
device(config)# vlan 1100
device(config-vlan-1100)#ingress-multicast
device(config-vlan-1100)#end
```

Syntax: [**no**] **ingress-multicast**

NOTE
Transparent-hw-flooding needs to be enabled before **ingress-replication** can be configured for a VLAN.

The following example shows the running VLAN after the command has been issued.

```
device(config)# show run vlan
no spanning-tree
!
!
vlan 1 name DEFAULT-VLAN
!
vlan 100
 transparent-hw-flooding
 ingress-multicast
!
```

The no form of the command will revert to the default mechanism (fabric plus egress replication) for this VLAN.

Syntax: [**no**] **ingress-multicast**

The following example configures three egress Traffic Managers that will be allowed per VLAN.

```
device# configure terminal
device(config)# vlan 1100
device(config-vlan-1100)#ingress-multicast max-tm 3
device(config-vlan-1100)#ingress-multicast
device(config-vlan-1100)#end
```

Syntax: ingress- multicast [max-tm *num***]**

The *num* parameter specifies the maximum number of egress Traffic Managers that will be allowed per VLAN. By default, the parameter is set to 4.

The following example shows the running VLAN after the command has been issued.

```
device(config)# show run vlan
no spanning-tree
!
!
vlan 1 name DEFAULT-VLAN
!
vlan 100
 transparent-hw-flooding
 ingress-multicast
!
vlan 1100
 transparent-hw-flooding
 ingress-multicast
 ingress-multicast max-tm 3
!
```

Ingress traffic shaping per multicast stream

Internet Protocol Television (IPTV) multicast streams on an individual inbound physical port are rate shaped to a specified rate and are prioritized over the broadcast or unknown-unicast traffic. Each IPTV multicast stream is queued separately and is scheduled independently to the outbound ports. The IPTV rate shaping reduces burstiness in the source stream.

NOTE
The number of active IPTV multicast streams for which per stream ingress shaping can be applied is limited to 512.

NOTE
Internet Protocol Television (IPTV) multicast streams are supported on Brocade NetIron CES and Brocade NetIron CER devices.

Implementation considerations

The considerations for implementing the multicast rate shaping are as follows:

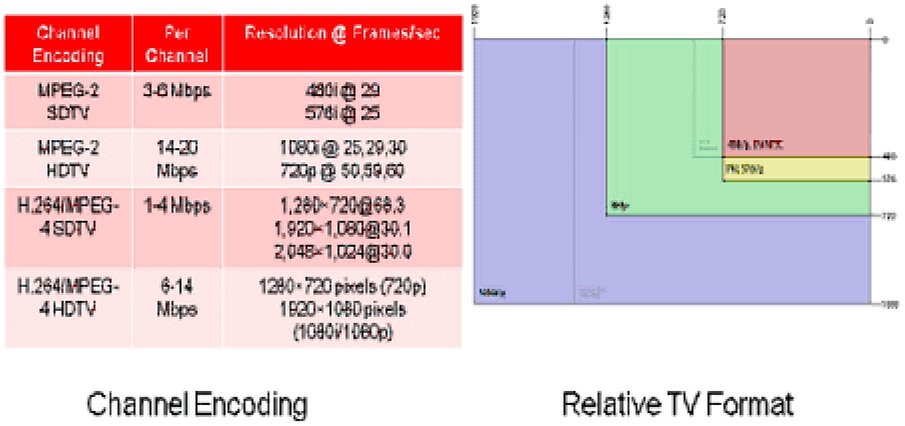
- At least the rate or the priority value must be specified.
- A maximum of 32 profiles and 64 profile-ACL bindings are allowed for multicast traffic.
- A single profile can be bound to multiple ACLs.
- An ACL can be associated only to one profile at a time.
- Either standard or extended ACLs can be used for multicast traffic shaping.
    - When a standard ACL is used, the address specified is treated as a group address and not as a source address.
    - When an extended ACL is used, the source and the destination addresses are treated as the source and group address of the multicast stream, respectively.

The IPTV Bandwidth Requirements figure below shows the type of IPTV channels and the bandwidth requirements for each type of channel using the following conventions:

- SDTV denotes Standard Definition Television
- HDTV denotes High Definition Television

NOTE
This feature is supported only on the 4x10 and 24x1 interface modules.

FIGURE 5 IPTV Bandwidth Requirements



Channel Encoding	Per Channel	Resolution @ Frames/sec
MPEG-2 SDTV	3-6 Mbps	480i @ 29 576i @ 25
MPEG-2 HDTV	14-20 Mbps	1080i @ 25,29,30 720p @ 50,59,60
H.264/MPEG-4 SDTV	1-4 Mbps	1,280×720@68.3 1,920×1,080@30.1 2,048×1,024@30.0
H.264/MPEG-4 HDTV	6-14 Mbps	1280×720 pixels (720p) 1920×1080 pixels (1080i/1080p)

Configuring multicast traffic policy maps

You can define profiles to match the IPTV multicast traffic of the individual ingress streams. To configure a policy map for the multicast streams, enter the following command.

```
device(config)# policy-map multicast sd_prof rate 2000 burst-size 2500 priority 2
queue-type 3
```

Syntax: [no] policy-map multicast *profile_name* [**rate** *r*] [**burst-size** *b*] [**priority** *0-7*] [**queue-type** *0-3*]

The *profile_name* is used to provide the parameters for traffic policing the multicast traffic.

The **rate** *r* variable specifies the shaping rate in bits per second. The value ranges from 100 kilobits per second (Kbps) through 20 gigabits per second (Gbps).

The **priority** *0-7* variable specifies the multicast traffic priority. The default value is four.

The **burst-size** *b* variable specifies the maximum number of bytes the multicast traffic is allowed to burst. The value ranges from 3 kilobytes (KB) through 128 KB. The default value is four KB.

The **queue-type** *0-3* variable specifies the queue type for which you want to set the priority. The default value is two. Optionally, you can specify the **burst-size** *b* and the **queue-type** *0-3* value to define a profile.

To delete a defined profile, enter the following command with the profile name.

```
device(config)# policy-map multicast sd_prof
```

The **no** form of the command resets the parameters to their default values.

NOTE
The rate and the priority value cannot be reset for a defined profile.

Binding multicast traffic policy maps

NOTE
A profile must exist in the configuration before it can be used for binding.

A standard or an extended ACL is used to define the IPTV streams that can be bound to a defined profile. The profile binding associates the properties of the profile to all the IPTV streams identified by the ACL. Binding of multicast streams can be done for Layer 3 multicast routing and Layer 2 multicast snooping.

Profile binding for Layer 3 multicast routing

You can bind a defined profile to a defined ACL for the default VRF or a specific VRF.

To bind the profile to the ACL in the default VRF, enter the following commands.

```
device(config)# ip multicast-routing policy-map r1 sd-1
device(config)# ipv6 multicast-routing policy-map r1q0 sdv61
```

To bind the profile to the ACL for a specified VRF, enter the following commands within the VRF "red" configuration context.

```
device(config)# ip vrf red
device(config-vrf-red)# address-family ipv4
device(config-vrf-red-ipv4)# ip multicast-routing policy-map r1 sd-1
device(config-vrf-red-ipv4)# exit-address-family
device(config-vrf-red)# exit-vrf
device(config)# ip vrf red
device(config-vrf-red)# address-family ipv6
device(config-vrf-red-ipv6)# ipv6 multicast-routing policy-map r1q0 sdv61
device(config-vrf-red-ipv6)# exit-address-family
device(config-vrf-red)# exit-vrf
```

Syntax: [no] ip multicast-routing policy-map *profile_name* {*acl_id* | *acl_name*}

The **ip multicast-routing policy-map** specifies ACL binding for IPv4 multicast routing.

The *profile_name* variable specifies the profile name of the multicast stream.

The *acl_id* | *acl_name* variable specifies the number and name of the standard ACL or an extended ACL. Enter a number from 1 through 99 for a standard ACL, and a number from 100 through 199 for an extended ACL.

Syntax: [no] ipv6 multicast-routing policy-map *profile_name* {*acl_id* | *acl_name*}

The **ipv6 multicast-routing policy-map** specifies ACL binding for IPv6 multicast routing.

The **no** form of the command removes the profile binding with the ACL in default VRF.

Profile binding for Layer 2 multicast snooping

You can bind a defined profile to a defined ACL per VLAN or per VPLS instance. To bind the profile to the ACL on VLAN 10, enter the following commands.

```
device(config)# vlan 10
device(config-vlan-10)# ip multicast policy-map r2q1 2
device(config-vlan-10)# ipv6 multicast policy-map r4q3 sdv64
```

Syntax: [no] ip multicast policy-map *profile_name* {*acl_id* | *acl_name*}

The **ip multicast policy-map** specifies the ACL binding for IPv4 multicast snooping.

Syntax: [no] ipv6 multicast policy-map *profile_name* {*acl_id* | *acl_name*}

The **ipv6 multicast policy-map** specifies the ACL binding for IPv6 multicast snooping.

The **no** form of the command removes the profile binding with the ACL on the VLAN or VPLS.

In the following example, binding for Layer 2 multicast snooping is applied to VPLS instance V1.

```
device(config)# router mpls
device(config-mpls)# vpls v1 10
device(config--mpls-vpls-v1)# multicast policy-map r2q1 2
device(config--mpls-vpls-v1)# multicast policy-map r4q3 sdv64
```

Syntax: [no] multicast policy-map *profile_name* {*acl_id* | *acl_name*}

NOTE
A profile that is bound cannot be deleted.

Configuration example for rate shaping IPTV multicast stream

The following example shows how to rate shape a multicast stream. In this example, to rate shape a multicast stream, profiles (sd_prof and hd_prof) are defined with rate and priority, ACLs (hd_streams and sd_streams) are configured which permit packets from four host IP addresses and denies all packets that are not explicitly permitted by the first four ACL entries, and then the profiles are bound to the defined ACLs.

```
device(config)# ip access-list standard hd_streams
device(config-std-nacl)# permit host 239.1.1.200
device(config-std-nacl)# permit host 239.1.1.201
device(config-std-nacl)# permit host 239.1.1.202
device(config-std-nacl)# permit host 239.1.1.203
device(config-std-nacl)# deny any
device(config-std-nacl)# exit

device(config)# ip access-list extended sd_streams
device(config-ext-nacl)# permit ip any 239.1.1.1/32
device(config-ext-nacl)# permit ip any 239.1.1.2/32
device(config-ext-nacl)# permit ip any 239.1.1.3/32
device(config-ext-nacl)# permit ip any 239.1.1.5/32
device(config-ext-nacl)# deny ip any any
device(config-ext-nacl)# exit

device(config)# policy-map multicast profile sd_prof rate 2000
device(config)# policy-map multicast profile hd_prof rate 14000 queue-type 2

device(config)# ip multicast-routing policy-map sd_prof sd_streams
device(config)# ip multicast-routing policy-map hd_prof hd_streams
```

Naming convention used in example

- sd_prof = Standard Definition profile
- hd_prof = High Definition profile
- sd_streams = Standard Definition TV stream
- hd_stream = High Definition TV stream

Traffic manager statistics display

Counters have been introduced to track the packets and bytes that enter the Ingress traffic manager and exit the egress traffic manager. Data from these counters can be displayed as described in the following sections.

Displaying all traffic manager statistics for a device

The following command displays all traffic manager statistics for a device by port groups that belong to each traffic manager.

```
device# show tm statistics
--------- Ports 2/1 - 2/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  464418
   EnQue Pkt Count:                          464418
   EnQue Byte Count:                         51904240
   DeQue Pkt Count:                          464418
   DeQue Byte Count:                         51904240
   TotalQue Discard Pkt Count:               0
   TotalQue Discard Byte Count:              0
   Oldest Discard Pkt Count:                 0
   Oldest Discard Byte Count:                0
Egress Counters:
   EnQue Pkt Count:                          701812
   EnQue Byte Count:                         78785888
   Discard Pkt Count:                        0
   Discard Byte Count:                       0
--------- Ports 4/1 - 4/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  0
   EnQue Pkt Count:                          0
   EnQue Byte Count:                         0
   DeQue Pkt Count:                          0
   DeQue Byte Count:                         0
   TotalQue Discard Pkt Count:               0
   TotalQue Discard Byte Count:              0
   Oldest Discard Pkt Count:                 0
   Oldest Discard Byte Count:                0
Egress Counters:
   EnQue Pkt Count:                          0
   EnQue Byte Count:                         0
   Discard Pkt Count:                        0
   Discard Byte Count:                       0
```

Syntax: show tm statistics

Displaying traffic manager statistics for a port group

The following command displays all traffic manager statistics for a specified port group as identified by a slot and port within the group.

```
device#show tm statistics ethernet 2/1
--------- Ports 2/1 - 2/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  464454
   EnQue Pkt Count:                          464454
   EnQue Byte Count:                         51907696
   DeQue Pkt Count:                          464454
   DeQue Byte Count:                         51907696
   TotalQue Discard Pkt Count:               0
   TotalQue Discard Byte Count:              0
   Oldest Discard Pkt Count:                 0
   Oldest Discard Byte Count:                0
Egress Counters:
   EnQue Pkt Count:                          701866
   EnQue Byte Count:                         78791072
   Discard Pkt Count:                        0
   Discard Byte Count:                       0
```

Syntax: show tm statistics ethernet *slot/port*

The *slot/port* variable specifies the slot and port number of the port group that you want to display traffic manager statistics for.

NOTE

A traffic manager contains a specific number of ports depending on the Interface module as described in the Traffic Manager statistics table in Displaying traffic manager statistics for an interface module on page 179. Specifying a particular port and slot gathers statistics for all ports that belong to the same port group.

Displaying traffic manager statistics for an interface module

The following command displays all traffic manager statistics for an interface module identified by its slot number.

```
device#show tm statistics slot 4
--------- Ports 4/1 - 4/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  0
   EnQue Pkt Count:                          0
   EnQue Byte Count:                         0
   DeQue Pkt Count:                          0
   DeQue Byte Count:                         0
   TotalQue Discard Pkt Count:               0
   TotalQue Discard Byte Count:              0
   Oldest Discard Pkt Count:                 0
   Oldest Discard Byte Count:                0
Egress Counters:
   EnQue Pkt Count:                          0
   EnQue Byte Count:                         0
   Discard Pkt Count:                        0
   Discard Byte Count:                       0
```

Syntax: show tm statistics ethernet *slot/port*

The **slot** *slot-number* variable specifies an interface module that you want to display traffic manager statistics from.

TABLE 34 Traffic manager statistics

This field...	Displays...
Ingress Statistics	
Total Ingress Pkt Count	A count of all packets entering into this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.
EnQue Pkt Count	A count of all packets entering Ingress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.
EnQue Byte Count	A count of all bytes entering Ingress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.
DeQue Pkt Count	A count of all packets dequeued from Ingress queues and forwarded on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.
DeQue Byte Count	A count of all bytes dequeued from Ingress queues and forwarded on this traffic manager.

TABLE 34 Traffic manager statistics (Continued)

This field...	Displays...
TotalQue Discard Pkt Count	A count of all packets failing to enter Ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering. A traffic manager contains a specific number of ports depending on the Interface module.
TotalQue Discard Byte Count	A count of all bytes failing to enter Ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering. A traffic manager contains a specific number of ports depending on the Interface module.
Oldest Discard Pkt Count	A count of all packets entering Ingress queues on this traffic manager, but deleted afterwards due to buffer full. A traffic manager contains a specific number of ports depending on the Interface module.
Oldest Discard Byte Count	A count of all bytes entering Ingress queues on this traffic manager, but deleted afterwards due to buffer full. A traffic manager contains a specific number of ports depending on the Interface module.
Egress statistics	
EnQue Pkt Count	A count of all packets entering egress queues and forwarded out on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.
EnQue Byte Count	A count of all bytes entering egress queues and forwarded out on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.
Discard Pkt Count	A count of all packets failing to enter egress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.
Discard Byte Count	A count of all bytes failing to enter egress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module.

NOTE
The byte counts displayed from the **show tm statistics** command incorporate proprietary internal headers of various lengths.

TABLE 35 Ethernet ports per traffic manager

Interface module	Ports per Traffic Manager (TM)	
	TM 1	TM 2

TABLE 35 Ethernet ports per traffic manager (Continued)

Interface module	Ports per Traffic Manager (TM)	
4 X 10 Gbps (Ethernet)	1 - 2	3 - 4
2 X 10 Gbps (Ethernet)	1 - 2	
20 X 1 Gbps (Ethernet)	1 - 20	

TABLE 36 Polling intervals for specified modules

Interface module	Interval (seconds)
4 X 10 Gbps (Ethernet)	180
2 X 10 Gbps (Ethernet)	180
20 X 1 Gbps (Ethernet)	180
8 X 10 Gbps (Ethernet)	180
2 X 100 Gbps (Ethernet)	1 80
4 X 40 Gbps (Ethernet)	1 80
24 x 10 Gbps (Ethernet)	600

Displaying traffic manager statistics for NI-MLX-10Gx8-M and NI-MLX-10Gx8-D modules

The following command displays traffic manager statistics for the NI-MLX-10Gx8-M module, and the NI-MLX-10Gx8-D module identified by its slot number.

```
device#show tm statistics slot 4

--------- Ports 4/1 - 4/4 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  61402830423
   EnQue Pkt Count:                          61402825288
   DeQue Pkt Count:                          61402692118
   TotalQue Discard Pkt Count:               5096
   Oldest Discard Pkt Count:                 0
Egress Counters:
   EnQue Pkt Count:                          95406035820
   Discard Pkt Count:                        0

--------- Ports 4/5 - 4/8 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  64207166485
   EnQue Pkt Count:                          64207161341
   DeQue Pkt Count:                          64207029336
   TotalQue Discard Pkt Count:               5087
   Oldest Discard Pkt Count:                 0
Egress Counters:
   EnQue Pkt Count:                          35018656764
   Discard Pkt Count:                        0
R4#
```

Syntax: show tm statistics [**slot** *slot-number*]

The **slot** *slot-number* variable specifies the slot number of the port group that you want to display traffic manager statistics for.

Displaying traffic manager statistics for the 4x10G module

The following command displays traffic manager statistics for the 4x10G module identified by its slot number.

```
device#show tm statistics slot 1

--------- Ports 1/1 - 1/2 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  37145922200
   EnQue Pkt Count:                          37145922200
   EnQue Byte Count:                         5943609079168
   DeQue Pkt Count:                          37145922200
   DeQue Byte Count:                         5943609079168
   TotalQue Discard Pkt Count:               0
   TotalQue Discard Byte Count:              0
   Oldest Discard Pkt Count:                 0
   Oldest Discard Byte Count:                0
Egress Counters:
   EnQue Pkt Count:                          83890318963
   EnQue Byte Count:                         13422682341696
   Discard Pkt Count:                        218
   Discard Byte Count:                       34752

--------- Ports 1/3 - 1/4 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  141547098478
   EnQue Pkt Count:                          141547098478
   EnQue Byte Count:                         22647526064544
   DeQue Pkt Count:                          141547098478
   DeQue Byte Count:                         22647526064544
   TotalQue Discard Pkt Count:               0
   TotalQue Discard Byte Count:              0
   Oldest Discard Pkt Count:                 0
   Oldest Discard Byte Count:                0
Egress Counters:
   EnQue Pkt Count:                          216769846687
   EnQue Byte Count:                         11606527206560
   Discard Pkt Count:                        0
   Discard Byte Count:                       0
```

Syntax: show tm statistics [**slot** *slot-number*]

The **slot** *slot-number* variable specifies the slot number of the port group that you want to display traffic manager statistics for.

Displaying traffic manager statistics for the 20x1G module

The following command displays traffic manager statistics for the 20x1G module.

```
device#show tm statistics all-counters 0
Ingress Counters:
   LBP Pkt Count:                            0
   QDP EnQue Pkt Count:                      0
   QDP EnQue Byte Count:                     0
   QDP DeQue Pkt Count:                      0
   QDP DeQue Byte Count:                     0
   QDP Head Delete Pkt Count:                0
   QDP Head Delete Byte Count:               0
   QDP Tail Delete Pkt Count:                0
   QDP Tail Delete Byte Count:               0
   Flow Status Message Count:                0
   Transmit Data Cell Count:                 0
   TDM_A Pkt Count:                          0
   TDM_B Pkt Count:                          0

Programmable Ingress Counters:
[Queue Select: 8000, Queue Mask 0x0007]
   QDP EnQue Pkt Count:                      0
   QDP EnQue Byte Count:                     0
   QDP DeQue Pkt Count:                      0
   QDP DeQue Byte Count:                     0
   QDP Head Delete Pkt Count:                0
   QDP Head Delete Byte Count:               0
   QDP Tail Delete Pkt Count:                0
   QDP Tail Delete Byte Count:               0
   Flow Status Message Count:                0

Egress Counters:
   EGQ EnQue Pkt Count:                      0
   EGQ EnQue Byte Count:                     0
   EGQ Discard Pkt Count:                    0
   EGQ Discard Byte Count:                   0
   EGQ Segment Error Count:                  0
   EGQ Fragment Error Count:                 0
   Port63 Error Pkt Count:                   0
   Pkt Header Error Pkt Count:               0
   Pkt Lost Due to Buffer Full Pkt Count:    0
   Reassem Err Discard Pkt Count:            0
   Reassem Err Discard Fragment(32B) Count:  0
   TDM_A Lost Pkt Count:                     0
   TDM_B Lost Pkt Count:                     0

Programmable Egress Counters:
[Port Id for Enque: 0 (Disable), Port Id for Discard: 0 (Disable)]
   EGQ EnQue Pkt Count:                      0
   EGQ EnQue Byte Count:                     0
   EGQ Discard Pkt Count:                    0
   EGQ Discard Byte Count:                   0
```

Syntax: show tm statistics all-counters *dev_id*

The *dev_id* variable specifies the device id that you want to display traffic manager statistics for.

Displaying traffic manager statistics for IPTV multicast queue

The following command displays traffic manager statistics for the IPTV Multicast queue on an Ethernet module.

```
device# show tm-voq-stat src_port eth 3/21 fid 8004
Multicast Queue-Id:8207
------------------------
Priority =  0/1
   EnQue Pkt Count                           8496
   EnQue Bytes Count                         11758464
   DeQue Pkt Count                           7743
   DeQue Bytes Count                         0
   Total Discard Pkt Count                   0
   Total Discard Bytes Count                 0
   Oldest Discard Pkt Count                  0
   Oldest Discard Bytes Count                0
   Current Queue Depth                       0
   Maximum Queue Depth since Last read       0
```

Syntax: show tm-voq-stat src_port eth *slot/port* **fid** *fid-id*

The **eth** *slot/port* parameter displays TM statistics for an individual port.

The **fid** *fid-id* parameter displays TM statistics for a specified fid.

TABLE 37 Output parameters of the show tm-voq-stat src_port command

Field	Description
Multicast Queue-Id	Shows the IPTV multicast queue identifier.
EnQue Pkt Count	Shows the count of all packets entering ingress queues on this traffic manager.
EnQue Bytes Count	Shows the count of all bytes entering ingress queues on this traffic manager.
DeQue Pkt Count	Shows the count of all packets dequeued from ingress queues and forwarded on this traffic manager.
DeQue Bytes Count	Shows the count of all bytes dequeued from ingress queues and forwarded on this traffic manager.
Total Discard Pkt Count	Shows the count of all packets failing to enter ingress queues on this traffic manager. This may be due to: • The queue reaching its maximum depth, WRED, or other reasons. • The network processor deciding to drop packets for reasons including: an unknown Layer 3 route, RPF, or segment filtering.
Total Discard Bytes Count	Shows the count of all bytes failing to enter ingress queues on this traffic manager. This may be due to: • The queue reaching its maximum depth, WRED, or other reasons. • The network processor deciding to drop packets for reasons including: an unknown Layer 3 route, RPF, or segment filtering.
Oldest Discard Pkt Count	Shows the count of all packets entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.
Oldest Discard Bytes Count	Shows the count of all bytes entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.
Current Queue Depth	Shows the current queue depth.
Maximum Queue Depth since Last read	Shows the maximum queue depth since last access to read.

Clearing traffic manager statistics

You can clear traffic manager statistics selectively for a specified port group, selectively for an interface module, or for an entire Brocade device as shown in the following.

```
device# clear tm statistics ethernet slot 4
```

Syntax: clear tm statistics [{**ethernet** *slot/port* | **slot** *slot-number*}]

Executing the **clear tm statistics** command without any options clears all traffic manager statistics on the device.

The **ethernet** *slot/port* option specifies a port group that you want to clear traffic manager statistics from.

The **slot** *slot-number* option specifies an interface module that you want to clear traffic manager statistics from.

New network processor counters displayed for packets to and from traffic manager

Output from the **show interface** command has been enhanced to provide the following traffic manager related information:

- Number of packets received at the network processor (NP)
- Number of packets sent from the NP to the traffic manager (TM)
- Number of Ingress packets dropped at the NP
- Number of packets transmitted from the NP
- Number of packets received by the NP from the TM

The following is an example of the new output from the show interface command with the changed output highlighted in **bold** .

```
device(config)# show interface ethernet 3/3
GigabitEthernet3/3 is up, line protocol is up
Hardware is GigabitEthernet, address is 0004.80a0.4052 (bia 0004.80a0.4052)
Configured speed auto, actual 1Gbit, configured duplex fdx, actual fdx
Member of L2 VLAN ID 1, port is untagged, port state is Forwarding
STP configured to ON, Priority is level0, flow control enabled
mirror disabled, monitor disabled
Not member of any active trunks
Not member of any configured trunks
No port name
MTU 1544 bytes, encapsulation ethernet
300 second input rate: 754303848 bits/sec, 1473249 packets/sec, 89.57% utilization
300 second output rate: 754304283 bits/sec, 1473250 packets/sec, 89.57% utilization
1015230949 packets input, 64974783168 bytes, 0 no buffer
Received 0 broadcasts, 0 multicasts, 1015230949 unicasts
0 input errors, 0 CRC, 0 frame, 0 ignored
0 runts, 0 giants
NP received 1039220106 packets, Sent to TM 1039220442 packets
NP Ingress dropped 0 packets
1015231660 packets output, 64974824768 bytes, 0 underruns
Transmitted 0 broadcasts, 0 multicasts, 1015231660 unicasts
0 output errors, 0 collisions
NP transmitted 1039221393 packets, Received from TM 1039221562 packets
```

QoS for NI-MLX-1Gx48-T modules

The NI-MLX-1Gx48-T module supports 48 1G port. In a fully loaded 32 slot chassis, there are only 8 queues supported on the TM port. The Brocade chassis supports 1008 ports with 8 queues per port. Beginning with this release, the Brocade configuration allows you configure more ports in the system by changing the TM port to use 4 queues instead of 8. The Brocade chassis supports 2016 ports using 4 queues per port.

Limitations on TM ports

The TM Port limitations are reached under the following situations.

1. When a new module type is configured.

2. When a new line card is inserted (no configured type).

3. When the user tries to configure the **max-tm-queue** parameter from 4 to 8.

The relationship between max TM queues and max TM ports are supported in the system as follows:

TABLE 38 Maximum TM queues and TM ports

Max TM queue per port	Max TM port
8	1008
4	2016

Configuring priority queues from 8 to 4

The **system-init max-tm-queues** command allows you to configure the maximum number of queues in TM to 4. To configure priority queues from 8 to 4, enter the following command.

```
device(config)# system-init max-tm-queues 4
```

Syntax: [no] system-init max-tm-queues *num*

The *num* value specifies the number of queues.

NOTE
When configuring priority queues from 8 to 4, or vice versa, the system displays the following message: **Reload required. Please write memory and then reload or power cycle. Failure to reload could cause system instability or failure.**

The NP continues to map all inbound packets to 8 internal priorities. If the **system-init max-tm-queues** command is configured, the NP will right shift this priority number by one bit before sending the packet to TM. The TM will en-queue the packets based on the following table:

TABLE 39 Queue type

NP priority	TM queue
7,6	3
5,4	2
3,2	1
1,0	0

Aggregated TM VOQ statistics collection

Supported modules

Traffic Manager queue statistics are only reported on the following interface modules:

- BR-MLX-10Gx8-X, NI-MLX-10Gx8-M, and NI-MLX-10Gx8-D
- BR-MLX-100Gx2-X and BR-MLX-100Gx1-X
- NI-X-OC192x2, NI-X-OC48x8, NI-X-OC48x4, and NI-X-OC48x2
- NI-MLX-48-T-A
- BR-MLX-24x1GF-X-ML, BR-MLX-24x1GC-X-ML, BR-MLX-24x1GF-X, and BR-MLX-24x1GC-X
- BR-MLX-10Gx24-DM (Added in NetIron 5.4.00b)

NOTE
The following modules are not supported NI-X-OC192x2, NI-X-OC48x8, NI-X-OC48x4, and NI-X-OC48x2.

Configuring aggregated TM VOQ statistics collection

When Traffic Manager (TM) Virtual Output Queue (VOQ) statistics collection is enabled, the system will start collecting TM queue statistics. To configure priority queues, refer to "Configuring priority queues from 8 to 4" on page 485 of the Brocade MLX Series and NetIron Family Configuration Guide.

Enabling aggregated TM VOQ statistics collection

The **tm-voq-collection** command allows you to enable and disable aggregated TM VOQ statistics collection.

```
device(conf
ig)# statistics
device(config-statistics)# tm-voq-collection
```

Syntax: no tm-voq-collection

NOTE
If priority queues are configured with system init max-tm-queues 4, TM will queue packets based on the values described in the table.

TABLE 40 Queue Type

NP priority	TM queue
7,6	3
5,4	2
3,2	1
1,0	0

Enabling SNMP support for brcdTMDestUcastQStatTable

Aggregated TM VOQ statistics per destination physical port per priority can be retrieved via SNMP using brcdTMDestUcastQStatTable. To enable SNMP agent support for brcdTMDestUcastQStatTable, use the **snmp-server enable mib tm-dest-qstat** command. By default, SNMP support for this table is disabled.

```
device(conf
ig)# snmp-server enable mib tm-dest-qstat
```

Syntax:[no] snmp-server enable mib tm-dest-qstat

NOTE
The **tm-voq-collection** command must be enabled along with the **snmp-server enable mib** command to enable SNMP support for aggregated TM VOQ statistics.

```
device(config)# statistics
device(config-statistics)# tm-voq-collection
```

Syntax:[no] tm-voq-collection

Enabling SNMP support for brcdNPQosStatTable

NP QOS statistics can be retrieved via SNMP using brcdNPQosStatTable. To enable SNMP agent support for brcdNPQosStatTable, use the **snmp-server enable mib np-qos-stat** command. By default, SNMP support for this table is disabled.

```
device(conf
ig)# snmp-server enable mib np-qos-stat
```

Syntax: [no] snmp-server enable mib np-qos-stat

NOTE
The **enable-qos-statistics** command must be enabled along with the **snmp-server enable mib np-qos-stat** command to enable SNMP support for retrieving NP QoS statistics.

NOTE
NP QOS statistics are supported for physical ports only.

```
device(config)# enable-qos-statistics
```

Syntax: [no] enable-qos-statistics

Displaying TM statistics from one queue or all queues

Use the following command to display traffic manager statistics for ethernet.

```
device# show tm-voq-stat src_port eth 2/1 dst_port ethernet
-----ethernet 2/2 - 1/4-----------------
EnQue Pkt Count                          4168645330
   EnQue Bytes Count                        1010575722
   DeQue Pkt Count                          0
   DeQue Bytes Count                        0
   Total Discard Pkt Count                  2084322665
   Total Discard Bytes Count                 505287857
   Oldest Discard Pkt Count                  0
   Oldest Discard Bytes Count                0
   WRED Dropped Pkt Count                    1594822490
   WRED Dropped Bytes Count                  126321962
   Current Queue Depth                       0
   Maximum Queue Depth since Last read       0
```

Use the following command to display traffic manager statistics for all priorities.

```
device# show tm-voq-stat   src_port  p1/1   dst_port p1/2
--------- Ports 1/1 - 1/4  ---------
Priority  =  0
   EnQue Pkt Count                                     81581531
   EnQue Bytes Count                               2692190523
   DeQue Pkt Count                                 81581531
   DeQue Bytes Count                               2692190523
   Total Discard Pkt Count                         0
   Total Discard Bytes Count                       0
   Oldest Discard Pkt Count                        0
   Oldest Discard Bytes Count                   0
   WRED Dropped Pkt Count                       0
   WRED Dropped Bytes Count                        0
   Current Queue Depth                                 0
   Maximum Queue Depth since Last read      2310
Priority =  1
   EnQue Pkt Count                                     0
   EnQue Bytes Count                                   0
   DeQue Pkt Count                                     62
   DeQue Bytes Count                                   1302
   Total Discard Pkt Count                         0
   Total Discard Bytes Count                       0
   Oldest Discard Pkt Count                        0
   Oldest Discard Bytes Count                      0
   WRED Dropped Pkt Count                       0
   WRED Dropped Bytes Count                      21
   Current Queue Depth                             0
   Maximum Queue Depth since Last read      0
Priority =  2
....
```

Syntax: show tm-voq-stat src_port *source-port* **dst_port ethernet** *destination-port* [*priority*]

Specification of a *source-port* and *destination-port* is required.

You can optionally specify a *priority* to limit the display to a single priority.

The output from the TM Q statistics is available only if the src card type is a module listed in the supported modules list

You can optionally specify a *priority* to limit the display to a single priority or use the **all** parameter to display all priorities.

TABLE 41 Traffic Manager statistics

This field...	Displays...
EnQue Pkt Count	A count of all packets entering ingress queues on this traffic manager.
EnQue Byte Count	A count of all bytes entering ingress queues on this traffic manager.
DeQue Pkt Count	A count of all packets dequeued from ingress queues and forwarded on this traffic manager.
DeQue Byte Count	A count of all bytes dequeued from ingress queues and forwarded on this traffic manager.

TABLE 41 Traffic Manager statistics (Continued)

This field...	Displays...
TotalQue Discard Pkt Count	A count of all packets failing to enter ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering.
TotalQue Discard Byte Count	A count of all bytes failing to enter ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering.
Oldest Discard Pkt Count	A count of all packets entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.
Oldest Discard Byte Count	A count of all bytes entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.
WRED Dropped Pkt Count	A count of all packets entering ingress queues on this traffic manager but dropped due to WRED.
WRED Dropped Bytes Count	A count of all bytes entering ingress queues on this traffic manager but dropped due to WRED.
Maximum Queue Depth since Last read	The maximum queue depth since last access to read.

Displaying TM statistics from the multicast queue

Use the following command to display traffic manager statistics from the Multicast queue for priority 1 on a module.

```
device# show tm-voq-stat src_port eth 4/1 multicast 1
Priority = 0/1
   EnQue Pkt Count                          0
   EnQue Bytes Count                        0
   DeQue Pkt Count                          0
   DeQue Bytes Count                        0
   Total Discard Pkt Count                  0
   Total Discard Bytes Count                0
   Oldest Discard Pkt Count                 0
   Oldest Discard Bytes Count               0
   WRED Dropped Pkt Count                   0
   WRED Dropped Bytes Count                 0
   Current Queue Depth                      0
   Maximum Queue Depth since Last read      0
```

Syntax: show tm-voq-stat src_port *source-port* **multicast** [{*priority* | **all**}]

Specification of a *source-port* is required.

You can optionally specify a *priority* to limit the display to a single priority or use the **all** parameter to display all priorities.

Showing collected aggregated TM VOQ statistics

Aggregated counters are the aggregation of counters from all ingress TMs to a specified destination port. Use the **show tm-voq-stats dst_port** command to display aggregated counters for each port.

```
device# show tm-voq-stat dst_port ethernet 8/1 4
   Port     Enqueued (pkts)        Dequeued (pkts)        Dropped (pkts)
             (bytes)                (bytes)                (bytes)
-------------------------------------------------------------------------------
8/1   :
P4    :                    1277236787
1377236787                                   0
                           198322097328
198322097328                                 0
```

Syntax: show tm-voq-stats dst_port ethernet {*slot/port* | **all** } { *priority* | **all** }

The *slot/port* variable specifies the slot and port number of the port group from which you want to display Traffic Manager statistics. The **all** option displays TM statistics for all ports.

The *priority* variable specifies the priority for which you want to display Traffic Manager statistics. The **all** option displays TM statistics for all priorities.

The following combinations of the command can be used to display Traffic Manager statistics:

- Use the **show tm-voq-stats dst_port ethernet** *slot/port P* command to display priority P counters for the slot/port.
- Use the **show tm-voq-stats dst_port ethernet** *slot/port* **all** command to display aggregated and all priorities counters for the *slot/port*.
- Use the **show tm-voq-stats dst_port ethernet** *slot/port* command to display aggregated counters for the *slot/port*.
- Use the **show tm-voq-stats dst_port all** *P* command to display priority *P* counters for all the ports in the system.
- Use the **show tm-voq-stats dst_port all** command to display aggregated counters for all the ports in the system.
- Use the **show tm-voq-stats dst_port all all** command to display all priorities and aggregate counters for all the ports in the system.

NOTE
The statistics are shown only when aggregated TM VOQ statistics collection is enabled.

TABLE 42 CLI display of show tm-voq-stats dst_port command

Field	Description
Enqueued (pkts) (bytes)	An aggregated count of all enqueued packets (and bytes) per egress port per priority.
Dequeued (pkts) (bytes)	An aggregated count of all dequeued packets (and bytes) per egress port per priority.
Dropped (pkts) (bytes)	An aggregated count of all dropped packets (and bytes) per egress port per priority.

Displaying TM VOQ depth summary

The **show tm-voq-stat max-queue-depth slot** command provides summary of the maximum queue depth of any queue from TM and provides additional information for debugging purposes. Knowing the maximum queue depth also allows a way to set maximum queue size adjusted for the specific traffic patterns on a system.

The following example displays summary of the maximum queue depth from the TM:

```
device#show tm-voq-stat max-queue-depth slot

--------- Ports 3/1 - 3/24 ---------
QType    Max Depth       Max Util        Destination Port
0        1013804         96%                    3/1
1        1013848         96%                    3/1
2        1013666         96%                    3/4
3        1013794         96%                    3/1
4        1013564         96%                    3/1
5        538             0%                     2/7
6        532             0%                     2/7
7        0               0%                     NA

--------- Ports 3/25 - 3/48 ---------
QType    Max Depth       Max Util        Destination Port
0        0               0%                     NA
1        0               0%                     NA
2        0               0%                     NA
3        0               0%                     NA
4        0               0%                     NA
5        0               0%                     NA
6        0               0%                     NA
7        0               0%                     NA
```

Syntax: show tm-voq-stat max-queue-depth slot *slot_number*

TABLE 43 TM voq maximum queue depth summary.

Field	Description
QType	It specifies the queue priority.
Max Depth	It specifies the maximum queue depth of any queue with Qtype in bytes.
Destination Port	It specifies the destination port of queue that had highest max queue depth.
Max Util	It specifies the percentage of maximum queue utilization (max-queue-depth / max-queue-size).

Displaying TM buffer utilization

The **show tm buffer-pool-stats slot** command displays the maximum buffer utilization from the TM and provides additional information for debugging purposes.

The following example displays maximum buffer utilization:

```
device#show tm buffer-pool-stats slot

--------- Ports 3/1 - 3/4 ---------
Maximum Buffer Size: 0 (0%)
Maximum Occupied Buffer Descriptors: 0 (0%)
--------- Ports 3/5 - 3/8 ---------
Maximum Buffer Size: 0 (0%)
Maximum Occupied Buffer Descriptors: 0 (0%)
```

Syntax: show tm buffer-pool-stats slot *slot_number*

TABLE 44 TM buffer pool statistics.

Field	Description
Maximum Buffer Size	Specifies the maximum buffer size in bytes for both gold and bronze traffic and also shows percentage of buffer used out of maximum packet buffer.
Maximum Occupied Buffer Descriptors	Specifies the maximum buffer pointers used for both gold and bronze traffic and also shows percentage of buffer pointers used out of total buffer points.

Displaying TM statistics for queue-drops

Use the following command to display traffic manager statistics to find all the source ports, whose traffic is resulting in non-zero drops for a specified destination port.

TABLE 45 Traffic Manager statistics for queue drops

This field...	Displays...
EnQue Pkt Count	A count of all packets entering ingress queues on this traffic manager.
EnQue Byte Count	A count of all bytes entering ingress queues on this traffic manager.
DeQue Pkt Count	A count of all packets dequeued from ingress queues and forwarded on this traffic manager.
DeQue Byte Count	A count of all bytes dequeued from ingress queues and forwarded on this traffic manager.
TotalQue Discard Pkt Count	A count of all packets failing to enter ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering.
TotalQue Discard Byte Count	A count of all bytes failing to enter ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering.

show tm-voq-stat queue-drops command has this output.

```
device# show tm-voq-stat queue-drops dst_port eth 4/1 all
Port     Enqueued (pkts)    Dequeued (pkts)    Dropped (pkts)
            (bytes)            (bytes)          (bytes)
------------------------------------------------------------------
2/(1-4)     567567            234234            333333
        66747623            23967890            42779733

P0:        234524            45645                88879
        54534534            35645665            18888869

P1:            0                    0                    0
                0                    0                    0
:
:
:
P7:            0                    0                    0
                0                    0                    0
```

Displaying TM statistics for destination LAGs

Use the following command to display traffic manager statistics for destination LAGs.

Syntax **show tm-voq-stat dst_lag** { *LAG_name* | *priority*|**all** }

Parameters **Lag_name**

Specifies the LAG name or number from which you want to display Traffic Manager statistics.

priority

Specifies the priority for which you want to display Traffic Manager statistics.

All

The All option displays TM statistics for all priorities.

Command Output The **show tm-voq-stat dst_lag** command displays the following information:

Field	Description
Enqueued (pkts) (bytes)	An aggregated count of all enqueued packets (and bytes) per egress port per priority.
Dequeued (pkts) (bytes)	An aggregated count of all dequeued packets (and bytes) per egress port per priority.
Dropped (pkts) (bytes)	An aggregated count of all dropped packets (and bytes) per egress port per priority.

Examples The following is an example of the output from the **show tm-voq-stat dst_lag <lag> <all>** command.

```
device#  show tm-voq-stat dst_lag lag1 all
 Port     Enqueued (pkts)         Dequeued (pkts)         Dropped (pkts)
                         (bytes)                 (bytes)                 (bytes)
--------------------------------------------------------------------------------
   LAG   :      12565554968             12390397203             1537204851
                1457604376288           1437286075548           178315762716

   P0    :      1200168136              1150090235              590854857
                139219503776            133410467260            68539163412

   P1    :      1248361118              1150092971              590844072
                144809889688            133410784636            68537912352

   P2    :      1443672053              1441685565              299253089
                167465958148            167235525540            34713358324

   P3    :      1740867220              1729494718              11461500
                201940597520            200621387288            1329534000

   P4    :      1740951743              1729475533              11480800
                201950402188            200619161828            1331772800

   P5    :      1730445747              1729455359              11500908
                200731706652            200616821644            1334105328

   P6    :      1730447997              1729457475              11498728
                200731967652            200617067100            1333852448

   P7    :      1730640954              1730645347              10310897
                200754350664            200754860252            1196064052
```

The following is an example of the **show tm-voq-stat dst_lag <lag> <priority>** command.

```
device# show tm-voq-stat dst_lag lag1 6
        Port     Enqueued (pkts)         Dequeued (pkts)         Dropped (pkts)
                         (bytes)                 (bytes)                 (bytes)
--------------------------------------------------------------------------------
   LAG   :
   P6    :      1825291617              1824301044              11498728
                211733827572            211618921104            1333852448
```

History

Release version	Command history
5.7.00	This command was introduced.
	This command was modified to...

Clearing the TM VOQ statistics

Use the **clear tm-voq-stat** command to clear all the counters for a specified port, or all ports, when the statistics collection is enabled. This command also clears SNMP statistics reported in brcdTMDestUcastQStatTable.

```
device(config)# clear tm-voq-stat dst_port all
```

Syntax: clear tm-voq-stat dst_port {**ethernet** *slot/port* | **all**} [*decimal* |**all**]

The *slot/port* variable specifies the slot and port number of the port group from which you want to clear traffic manager statistics. The **all** option clears traffic manager statistics from all ports.

The *decimal* variable specifies the priority from which you want to clear traffic manager statistics. The **all** option clears all priorities from the selected ports.

Clearing TM VOQ depth summary

Use the **clear tm-voq-stat max-queue-depth slot** command to clear the maximum queue depth summary of any queue.

```
device#clear tm-voq-stat max-queue-depth slot
```

Syntax: clear tm-voq-stat max-queue-depth slot *slot_number*

The *slot_number* specifies the decimal value of the slot number of the group from which you want to clear traffic manager queue depth summary.

Clearing TM buffer utilization

Use the **clear tm buffer-pool-stats slot** command to clear the maximum buffer utilization from TM.

```
device#clear tm buffer-pool-stats slot
```

Syntax: clear tm buffer-pool-stats slot *slot_number*

The *slot_number* specifies the decimal value of the slot number of the group from which you want to clear traffic manager buffer utilization.

Clearing queue-drops statistics

The clear queue-drops command is used to clear only the clear-drops statistics from the TM VOQ statistics for a specified port, or all ports, when the statistics collection is enabled.

```
device(config)# clear tm-voq-stat queue_drops dst_port ethernet
```

Syntax: clear tm-voq-stat queue_drops {**dst_port ethernet** *slot/port*} {*priority*| **all**}

The *slot/port* variable specifies the slot and port number of the port group from which you want to clear queue_drops statistics. The **all** option clears queue_drops statistics from all ports.

The *priority* variable specifies the priority from which you want to clear queue_drops statistics. The **all** option clears queue_drops statistics from all priorities from the selected ports.

Clearing TM destination LAG statistics

The clear tm-voq-stat dst_lag command is used to clear only the destination LAG statistics from the TM VOQ statistics for a specified LAG, or priority, when the statistics collection is enabled.

The following is an example of the clear tm-voq-stat dst_lag command.

```
device(config)#clear tm-voq-stat dst_lag LAG1
```

Syntax: clear tm-voq-stat dst_lag {**dst_lag** *lag_name*} {*priority*| **all**}

The *lag_name* variable specifies the LAG name from which you want to clear dst_lag statistics.

The *priority* variable specifies the priority from which you want to clear queue_drops statistics. The **all** option clears dst_lag statistics from all priorities from the selected ports.

Displaying QoS packet and byte counters

You can enable the collection of statistics for Ingress and Egress packet priorities using the **enable-qos-statistics** command. Once the collection of statistics is enabled, the **show np statistics** command can be used to display a count of the packet priorities of Ingress and Egress packets as shown in the following.

```
device# show np statistics
TD: Traffic Despritor. Each TD has size of 512 Bytes

MODULE # 0 PPCR # 0 :
Ingress Counters :
Received packets                                      = 5172
Discarded packets                                     = 0
Received TDs on traffic class 0                       = 0
Received TDs on traffic class 1                       = 0
Received TDs on traffic class 2                       = 0
Received TDs on traffic class 3                       = 0
Received TDs on traffic class 4                       = 0
Received TDs on traffic class 5                       = 0
Received TDs on traffic class 6                       = 0
Received TDs on traffic class 7                       = 10344

Egress Counters :
Transmitted unicast packets                           = 0
Transmitted multicast packets                         = 0
Transmitted broadcast packets                         = 0
Filtered packets due to VLAN spanning tree            = 0
Tail dropped packets                                  = 0
```

```
Control packets                                      = 10344
Packets filtered due to egress forward restrictions   = 0
Packets dropped due to full multicast egress queue    = 91459

TD: Traffic Despritor. Each TD has size of 512 Bytes

MODULE # 1 PPCR # 0 :
Ingress Counters :
Received packets                                      = 47809289718
Discarded packets                                     = 0
Received TDs on traffic class 0                       = 47809289569
Received TDs on traffic class 1                       = 0
Received TDs on traffic class 2                       = 0
Received TDs on traffic class 3                       = 0
Received TDs on traffic class 4                       = 0
Received TDs on traffic class 5                       = 0
Received TDs on traffic class 6                       = 0
Received TDs on traffic class 7                       = 0

Egress Counters :
Transmitted unicast packets                           = 18561287821
Transmitted multicast packets                         = 0
Transmitted broadcast packets                         = 0
Filtered packets due to VLAN spanning tree            = 0
Tail dropped packets                                  = 5910551222
Control packets                                       = 0
Packets filtered due to egress forward restrictions   = 0
```

Packets dropped due to full multicast egress queue = 0

QoS commands affected by priority queues

- Priority-based Rate Shaping
- Weighted Random Early Discard (WRED)
- Weighted-based Scheduling and Mixed Strict Priority
- CPU Copy Queue
- Traffic Manager Statistics

Priority-based rate shaping

If the user specifies a priority of 4-7 when the max-tm-queues parameter is configured using 4 queues, the **qos shaper priority** command is accepted, but a warning message is displayed.

The following example displays the **qos shaper priority** command configured with a priority 4 shaper.

```
device(config-if-eth-16/1)# qos shaper priority 4 1000000
Warn: current max TM queues is 4-configuration of priority 4-7 will not have any
effect.
```

NOTE
If a priority 5 shaper is already configured, and the max-tm-queues parameter changes from 8 to 4 queues and is reloaded, then the priority 5 shaper configuration line is still displayed. The priority shaper 5 will not take effect.

Weighted Random Early Discard (WRED)

When WRED is enabled for a queue type of any forwarding queue, it will receive a warning message that is similar to when the **priority-based rate shaping** command is configured. Refer to Priority-based rate shaping on page 198 for more information.

The following example displays enabling WRED for the forwarding queues with a queue type of 6.

```
device(config)#qos queue-type 6 wred enable
Warn: current max TM queues is 4-configuration of queue-type 4-7 will not have any
effect.
```

NOTE
If the **system-init max-tm-queues 4** command is configured, the user is able to configure similar WRED parameters, such as Average weight, Max Instantaneous queue size, Drop Precedence, etc. for all priorities. The default values of all WRED parameters (refer to the Queue type table in Configuring priority queues from 8 to 4 on page 186) is only effective when queue-type 0-3 is used.

Weighted-based scheduling and mixed strict priority

When the **max-tm-queues** parameter is configured with 8 or 4 queues, the **qos scheduler weighted** command and **qos scheduler mixed** command will still take the same number of weight values, but the unnecessary priority values are ignored.

The following example displays when the **qos scheduler weighted** command is configured using 4 queues.

```
device(config-ethe-1/1)#qos scheduler weighted 7 6 5 4 3 2 11
Current max TM queues is 4 - weights "7", "6", "5", "4" for priority 7-4 will not
have any effect.
```

The following example displays when the **qos scheduler mixed** command is configured using 4 queues.

```
device(config-ethe-1/1)#qos scheduler mixed 4 3 2 1 1
Current max TM queues is 4 - weights "4", "3", "2" for queues 4-2 will not have any
effect.
```

The following table displays how traffic scheduling for Strict Priority-based Scheduling and Weighted-based Scheduling is configured differently between 8 and 4 queues:

TABLE 46 Strict v.s. weighted queues (Continued)

	8 queues (current)	4 queues (new)
Strict	7,6,5	3,2
Weighted	4,3,2,1,0	1,0

Error messages for CPU copy queue and traffic manager statistics

The following error messages are displayed for CPU copy queue and traffic manager statistics when the incorrect queues are configured.

CPU copy queue

When **system-init max-tm-queues 4** command is configured, the **rl-cpu-copy** command displays a warning message when the user specifies a priority 4-7.

```
device(config)# rl-cpu-copy priority 4 1000000
Warn: current max TM queues is 4-configuration of priority 4-7 will not have any
effect.
```

Traffic manager statistics

When **system-init max-tm-queues 4** command is configured, the **show tm-voq-stat** command will only take a priority 0-3. An error message is displayed when an invalid priority range is enabled.

```
device#show tm-voq-stat src_port eth 9/1 dst_port ethernet 2/3 5
Error: priority range 0 to 3.
```

NOTE
The **show tm-voq-stat** command will print statistics for 4 queues, instead of 8. The output from the TM Q statistics is available only if the src card type is a 48x1GC module or 8x10G module.

Enhanced buffer management for NI-MLX-10Gx8 modules and NI-X-100Gx2 modules

The prioritized buffer-pool feature establishes two buffer-pools, gold buffer-pool for high priority traffic and bronze buffer-pool for low priority traffic. Each internal priority can be associated with either the gold buffer-pool or the bronze buffer-pool. High priority traffic is guaranteed buffers even in the presence of bursty low priority traffic. This is only applicable to 8x10G and 2x100G modules.

The Virtual Output Queue (VOQ) size configuration enables the user to increase the queue size.

Enhanced Packet Buffer Management

Two buffer-pools are established, gold buffer-pool for high priority traffic and bronze buffer-pool for low priority traffic. Each internal priority can be associated with either the gold buffer-pool or the bronze buffer-pool. This guarantees buffers for high priority traffic, even in the presence of bursty low priority traffic. By default the internal priority/queue-type 7 is associated with the gold buffer-pool and the internal priorities/queue-types 6-0 are associated with the bronze buffer-pool.

Enhanced Packet Buffering Considerations

- For the 8x10G family, the buffer-pools apply only to unicast forwarded traffic for the 8x10G family.
- For the 2x100G family, the buffer-pools are common for unicast/unknown-unicast/broadcast/multicast forwarded traffic.

The buffer-pools are calculated as follows: "Gold buffer-pool minimum guarantee percentage = 100% - Bronze buffer-pool percentage" and "Bronze buffer-pool minimum guarantee percentage = 100% -

Gold buffer-pool percentage". If the minimum buffer guarantee percentage is less than "0", then the minimum buffer guarantee percentage is "0".

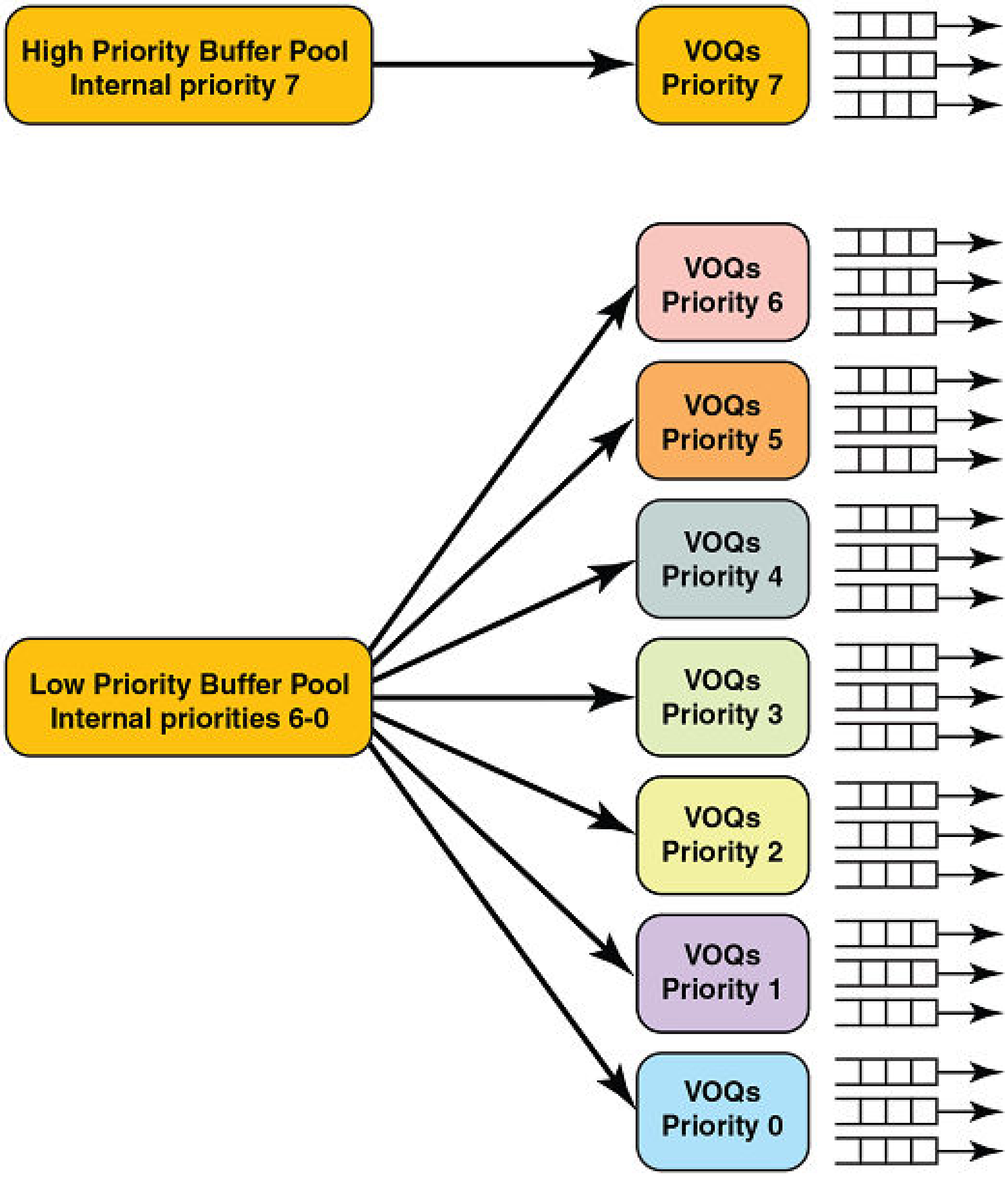
Default Configuration

For the 8x10 family of modules provides a minimum buffer guarantee for unicast control protocol traffic. The Gold buffer-pool has 100% of physical memory and Bronze buffer-pool has 95% of physical memory.

For 2x100G module - The goal is to provide a minimum buffer guarantee for unicast/multicast control protocol traffic. The buffer-pool is shared between Unicast/Broadcast/Unknown-unicast/Multicast. The Gold buffer-pool has 100% of physical memory and Bronze buffer-pool has 95% of physical memory.

The system default is depicted in Figure 1 and Figure 2.

FIGURE 6 System Default Priorities and Corresponding Buffer Pools



Configuration of buffer-pool priority to queue type

The following table displays the traffic types that are associate with each priority. Buffer-pool configuration enables the mapping of priorities to either the gold or bronze buffer-pool. VOQ Priority 7 cannot be removed from the Gold buffer-pool. VOQ Priority 0 cannot be removed from the Bronze buffer-pool.

TABLE 47 Strict v.s. weighted queues

Queue Type (Prioritization Category)	Protocol
P7	LACP, UDLD (802.3ah), STP/RSTP/BPDU, VSRP, MRP, BFD, GRE-KA/IS-IS over GRE, G.8032, LLDP, non-CCM 802.1ag (Eth + MPLS-enc.), BFD(Single-hop/Multi-hop/MPLS), IS-IS Hello, OSPFv2 Hello(GRE+Eth), OSPFv3 Hello(6to4+eth)
P6	IS-IS Non-Hello, OSPFv2/3 Non-Hello (Eth, GRE, 6to4), IPSec ESP Packets (for OSPFv3 over IP-sec), OSPF/OSPF over GRE or 6to4, IS-IS, RIP/RIPNG, VRRP (v4/v6), VRRPE(v4/v6)
P5	BGP/BGP over GRE or 6to4, LDP (basic and extended), RSVP, CCP(MCT), 802.1ag CCM (Eth + MPLS-enc.)
P4	PIM/PIM over GRE or 6to4, VPLS Encapsulated PIM, MSDP/MSDP over GRE/MSDP over VPLS
P3	IGMP, VPLS Encapsulated IGMP, GRE encapsulated IGMP, ARP, MLD, DHCP/BOOTP, ND6/ND6 in 6to4
P2	IPv4 Router Alert
P1	New Unassigned Protocols
P0	Existing Unassigned Protocols: GARP, ESIS, L2-Trace, REMOVE ESIS Prioritization

You can map each queue type to the buffer-pool type as needed. The queue-types are individually selected to be placed in the desired buffer-pool.

To map a queue-type to a buffer-pool enter the following command:

```
device(config)#qos queue-type 5 buffer-pool bronze
```

Syntax: [no] qos queue-type *queue-number* **buffer-pool** {**gold** | **bronze**}

The *queue-number* variable signifies the queue-type priority which will be associated with the buffer-pool. The *queue-number* can vary from 1-6.

The **buffer-pool** option indicates the type of buffer-pool, gold or bronze.

Note: This command applies only to Unicast forwarded traffic. This does not apply to Broadcast/Unknown-unicast/Multicast forwarded traffic.

Configuration considerations

- Queue-type/internal priority 7 cannot be removed from Gold buffer-pool.
- Queue-type/internal priority 0 cannot be removed from Bronze buffer-pool.
- The command is applicable only to 8x10 family and 2x100G.
- This command is available only on switch reload.

Configuring buffer-pool size

```
device(config)#qos buffer-pool bronze 50
```

Syntax: [no] qos buffer-pool {gold | bronze} max-percentage

The **buffer-pool** indicates the type of buffer-pool, gold or bronze.

The *percentage* specifies the maximum percentage of memory allocated for the **buffer-pool** type.

Configuration considerations

- The percentage of memory allocated for Bronze buffer-pool cannot exceed 95%.
- The percentage of memory allocated for Bronze buffer-pool cannot be below 5%.
- The command is applicable only to 8x10 family and 2x100G.
- Check the configuration as priorities are placed into buffer-pools individually.

Configuration examples

The example below depicts where internal priorities 7-6 are associated with the Gold buffer-pool and internal priorities 5-0 with the Bronze buffer-pool.

```
device(config)# qos queue-type 6 buffer-pool gold
```

The example below depicts where the Bronze buffer-pool is configured with 70% of physical memory.

```
device(config)# qos buffer-pool bronze 70
```

Displaying buffer-pool information

To display the buffer-pool configuration enter the command:

```
device# show qos buffer-pool
Unicast Queue-Type        Buffer Type
                0                BRONZE
                1                BRONZE
                2                BRONZE
                3                BRONZE
                4                BRONZE
                5                BRONZE
                6                BRONZE
                7                GOLD

Multicast Queue-Type        Buffer Type
                3                       GOLD
                2                       BRONZE
                1                       BRONZE
                0                       BRONZE

Buffer Type         Memory(%)     Min. Guarantee(%)
BRONZE                95                 0
GOLD                  100                 5

Module Type  Total Memory   Max. Gold   Min. Gold   Max. Bronze   Min. Bronze
8x10              1392 MB    1392 MB      69 MB      1322 MB         0 MB
2x100             1472 MB    1472 MB      73 MB      1398 MB         0 MB
24x10             1472 MB    1472 MB      73 MB      1398 MB         0 MB
4x40              1472 MB    1472 MB      73 MB      1398 MB         0 MB
2X100-CFP2        1472 MB    1472 MB      73 MB      1398 MB         0 MB
20x10/1           1472 MB    1472 MB      73 MB      1398 MB         0 MB
```

Syntax: show qos buffer-pool

Configuring Virtual Output Queue (VOQ) queue size

Modules with 256MB VOQ size support

The following modules support a maximum VOQ size of 256MB. You also can disable the maximum VOQ size to allow the VOQ to grow to the size of the buffer-pool which the internal priority is associated with. This is recommended to be used only when there are two separate buffer-pools - one for high priority traffic and another for low priority traffic.

- NI-MLX-10Gx8-D
- NI-MLX-10Gx8-M
- NI-MLX-10Gx8-X
- NI-X-100Gx2

Configuration

To configure the max queue size for a queue-type (internal priority) enter the following command:

```
device(config)# qos queue-type 1 max-queue-size 256000
```

Syntax: [no] qos queue-type *queue-number* **max-queue-size** *max-queue*

The *queue-number* variable signifies the queue-type priority for which "max-queue-size" is changed. The *queue-number* can vary from 0-7.

The *max-queue* variable signifies the maximum value of the queue size in KB for the queue-type or internal priority. The *max-queue* can vary from 1-262144 KBytes.

Modules with 64MB VOQ size support

The following modules support a maximum VOQ size of 64MB.

- NI-MLX-X-10Gx4
- NI-MLX-X-1Gx24-GC
- NI-MLX-X-1Gx24-SPF
- NI-MLX-1Gx48-T

Configuration

To configure the max queue size for a queue-type (internal priority) enter the following command:

```
device(config)# qos queue-type 1 max-queue-size 64000
```

Syntax:[no] qos queue-type *queue-number* **max-queue-size** *max-queue*

The *queue-number* variable signifies the queue-type priority for which "max-queue-size" is changed. The *queue-number* can vary from 0-7.

The *max-queue* variable signifies the maximum value of the queue size in KB for the queue-type/ internal priority. The *max-queue* can vary from 0-65536 KBytes. Setting the "max-queue-size" to "0" implicitly sets the "max-queue-size" to the maximum value.

Legacy Modules VOQ size support

Legacy modules include module older that those listed above. Legacy modules support a maximum VOQ queue size of 32MB.

Configuration

To configure the max queue size for a queue-type (internal priority) enter the following command:

```
device(config)#qos queue-type 1 max-queue-size 15000
```

Syntax: [no] qos queue-type *queue-number* **max-queue-size** *max-queue*

The *queue-number* variable signifies the queue-type priority for which "max-queue-size" is changed. The *queue-number* can vary from 0-7.

The *max-queue* variable signifies the maximum value of the queue size in KB for the queue-type/ internal priority. The *max-queue* can vary from 0-32768 KBytes. Setting the **max-queue-size** to 0 implicitly sets the **max-queue-size** to the maximum value.

Hierarchical Quality of Service (HQoS)

TABLE 48 Supported Brocade Hierarchical QoS (HQoS) features

Features supported	Brocade NetIron XMR Series	Brocade NetIron MLX Series	Brocade NetIron CES 2000 Series BASE package	Brocade NetIron CES 2000 Series ME_PREM package	Brocade NetIron CES 2000 Series L3_PREM package	Brocade NetIron CER 2000 Series Base package	Brocade NetIron CER 2000 Series Advanced Services package
HQoS	Yes	Yes	No	No	No	No	No
PBB	Yes	Yes	No	No	No	No	No
Local VPLS	Yes	Yes	No	No	No	No	No
VPLS	No	No	No	No	No	No	No
IPv4	No	No	No	No	No	No	No
IPv6	No	No	No	No	No	No	No
Hierarchical Levels			No	No	No	No	No
10GX8-M Module	Yes	Yes	No	No	No	No	No
10GX8-X Module	Yes	Yes	No	No	No	No	No
10GX8-D Module	No	No	No	No	No	No	No
10Gx24 Module	No	No	No	No	No	No	No
HQoS Support for VPLS and LAG Interfaces	Yes	Yes	No	No	No	No	No
WRED Queue Management for HQoS	Yes	Yes	No	No	No	No	No

- Hierarchical QoS (HQoS) for 8x10G modules on page 208
- How HQoS works on page 208
- HQoS for Local VPLS on page 212
- HQoS for PBB traffic on page 213
- Bypassing hierarchy levels on page 214
- Configuring HQoS on page 216

Hierarchical QoS (HQoS) for 8x10G modules

NOTE
HQoS is supported on the egress of 10G ports of the NI-MLX-10GX8-M and BR-MLX-10GX8-Xmodules.HQoS is not supported on the NI-MLX-10GX8-D module.

Hierarchical QoS (HQoS) allows a carrier to consolidate different services on the same physical device running on the same physical infrastructure.

HQoS is a valuable tool, especially for networks that support multiple business customers who are running multiple applications with different prioritization and scheduling requirements over the same infrastructure.

HQoS uses an advanced scheduling mechanism, with multiple levels and multiple instances of scheduling and traffic shaping for the different services over a same connection. HQoS allows lower priority traffic to fully utilize the available bandwidth on a port, while ensuring high levels of QoS, e.g., low latency and guaranteed bandwidth, to higher priority traffic classes on that port. In summary, HQoS allows providers to offer improved customer SLAs and optimizes use of network resources.

How HQoS works

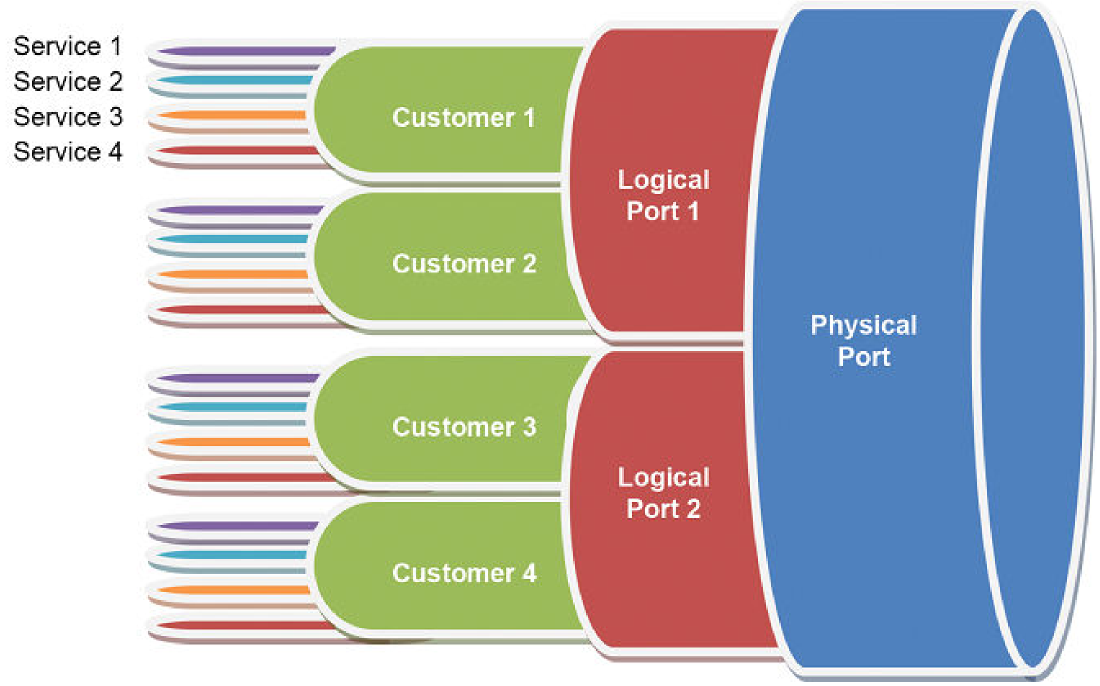
Hierarchical Quality of Service (HQoS) organizes a scheduler policy into a hierarchical tree that consists of a root node, branches nodes, and leaf nodes, where:

- The root node is the convergence point for all traffic and corresponds to a scheduler followed by a traffic shaper. The root node schedules and shapes the aggregated egress traffic of a physical port.
- A branch node is located in the middle of the hierarchy and corresponds to a scheduler followed by a traffic shaper.
- A leaf node corresponds to a scheduling queue.

HQoS scheduling levels do not support packet field matching capabilities. Packets are inspected once before being queued. Once packets go into a queue, everything beyond that point is a sequence of rate shapers and schedulers as defined by the hierarchical scheduling tree for that egress port.

HQoS supports a number of scheduling and shaping levels. Each level performs scheduling and shaping functions. By careful configuration, the different HQoS levels can map directly to a desired hierarchical traffic management model. For example, a hierarchy can be configured where an HQoS level represents the aggregated traffic of individual Customers, another level the individual VLANs of each customer, and another level the traffic following a Logical port downstream.

FIGURE 7 HQoS model



HQoS Components

The following HQoS components are supported in this release.

Supported levels of scheduling

At every scheduling/shaping level, the sum of the shaping rates going into a scheduler element does not need to add up to less than the shaping rate out of that scheduler element. The scheduler scheme used, Strict Priority (SP), Round Robin (RR), Weighted Round Robin (WRR), or mixed scheduling schemes, will determined how much traffic is scheduled for transmission from each scheduler flow. The rate shaper of a scheduler flow will limit the amount of traffic that can be scheduled from that scheduler flow. Therefore, the combination of scheduler flow rate shapers and scheduler element allows for the sharing of bandwidth among the respective scheduler flows in an ordered and pre-determined fashion.

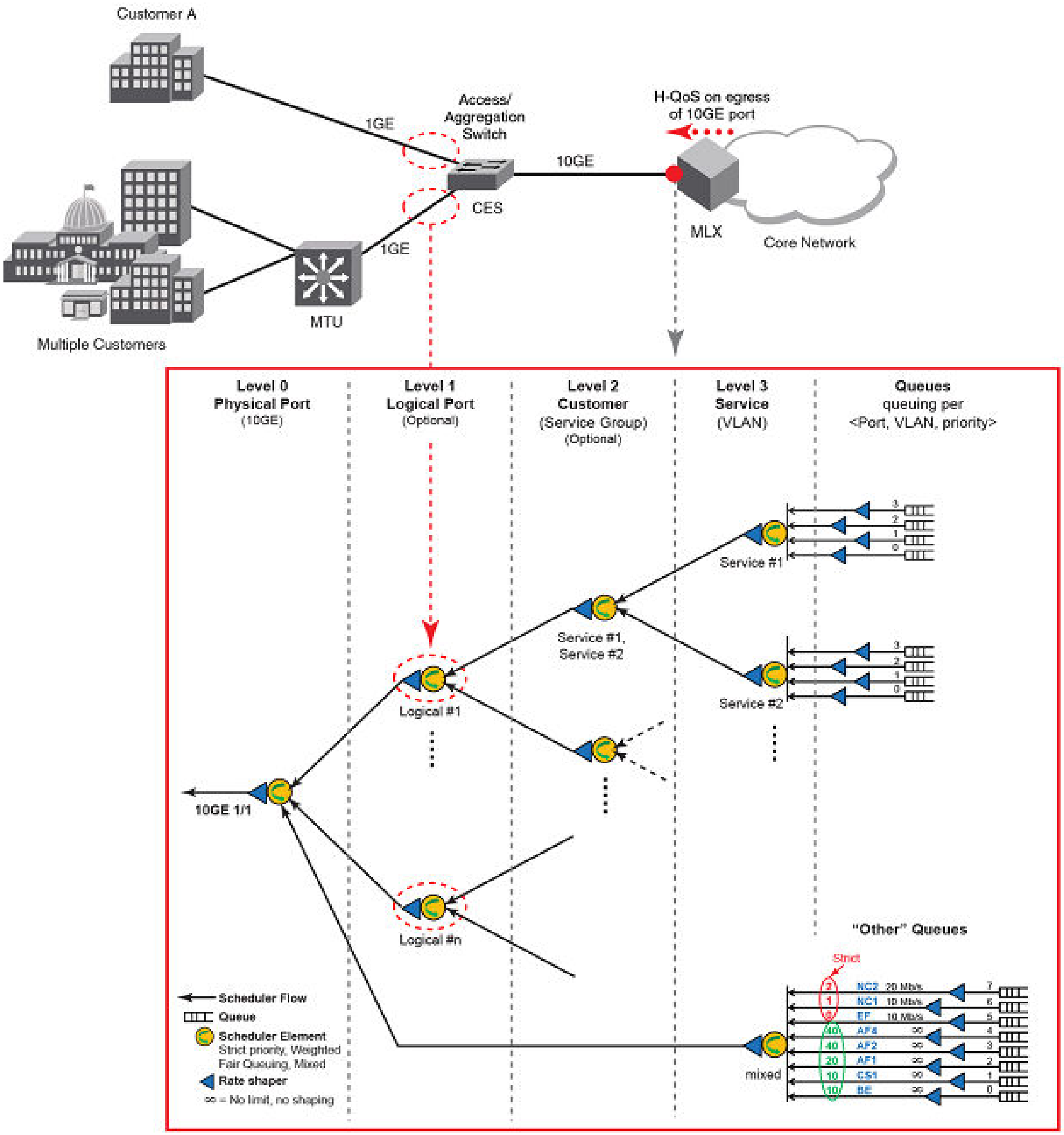
HQoS towards the customers

HQoS can shape the traffic towards the downstream 1GE links using the "Logical" port level of HQoS on Brocade devices.

In the following figure, two logical ports would be defined and shaped to 1Gb/s.

The HQoS policy is configured with Customer traffic connected to the appropriate "Logical" port on the HQoS hierarchy.

FIGURE 8 HQoS towards the Customers

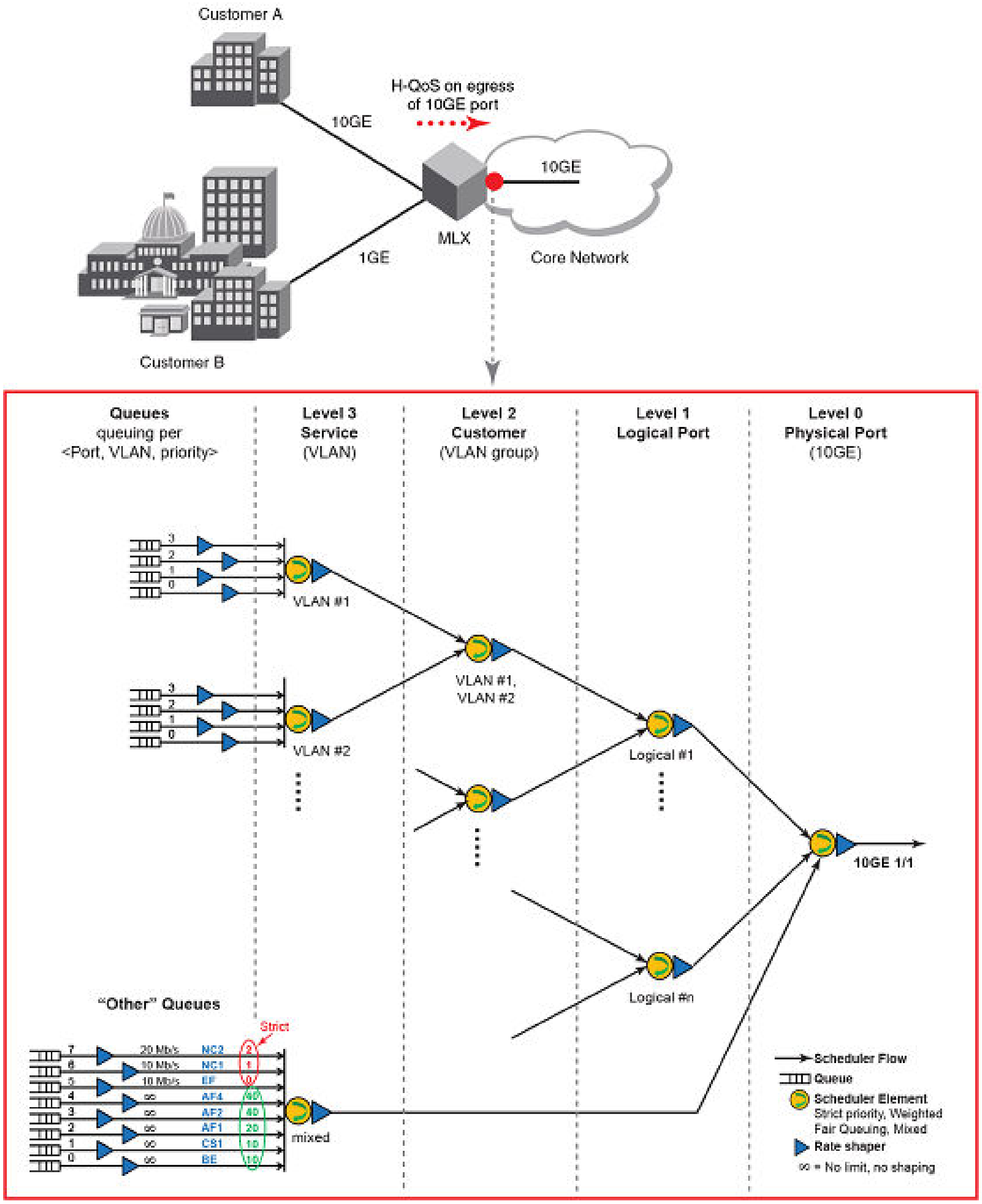


HQoS towards core network

HQoS usage towards the Service Provider core network on 10GE ports ensure high levels of QoS higher priority traffic classes for the customer.

The core network in this case can be a PB or PBB network.

FIGURE 9 HQoS towards Core Network



- **Level 3:** The Service level provides the scheduler/shaper for individual customer services, e.g., VLAN. The SLA applied here would apply to the individual service for that customer. For example, a customer may have two distinct point-to-point services identified by a SVLAN sharing the same physical link to the customer. The Service level schedules traffic from individual priority levels for a particular SVLAN. Priority levels may be served in Strict Priority (SP), Round Robin (RR), Fair Queuing (FQ), Weighted Round Robin (WRR), or mixed scheduling schemes.
- **Level 2:** The Customer level provides the scheduler/shaper for the aggregated services of a customer. For example, if a customer has two VLAN services at Level 3, this level would provide for scheduling/shaping of the combined traffic of the two customer VLANs. VLANs of a same customer

are commonly served in Round Robin (RR), Fair Queuing (FQ), or Weighted Round Robin (WRR) mode.

- **Level 1:** The Logical Port level provides schedulers/shapers for the traffic that would flow through the egress port of a downstream device. This level allows for scheduling and shaping of traffic to fit a downstream port. Customer scheduler flows are commonly served in Round Robin (RR), Fair Queuing (FQ), or Weighted Round Robin (WRR) mode.
- **Level 0:** The Physical Port level provides a scheduler/shaper for the egress port. The Physical Port level schedules traffic from individual Logical Port scheduler flows. Logical Port scheduler flows are commonly served in Round Robin (RR), Fair Queuing (FQ), or Weighted Round Robin (WRR) mode.

Scheduling traffic for forwarding

If the traffic being processed by a Brocade device is within the capacity of the device, all traffic is forwarded as received. Once it reaches the point where the device is bandwidth constrained, it becomes subject to drop priority if configured.

The Brocade devices classify packets into one of eight internal priorities. Traffic scheduling allows you to selectively forward traffic according to one of the following schemes:

- **Strict priority-based scheduling** - This scheme guarantees that higher-priority traffic is always serviced before lower priority traffic. The disadvantage of strict priority-based scheduling is that lower-priority traffic can be starved.
- **WFQ weight-based traffic scheduling** - This scheme services different traffic priorities based on defined weights. Available bandwidth is divided among the different traffic priorities proportionally to the defined weights. For example, two priority levels with a same assigned weight will be served with about the same bandwidth. A priority level with a weight value twice the weight value of another priority level will be served at about twice the bandwidth of the other priority level.
- **Mixed strict priority and weight-based scheduling** - This scheme provides a mixture of strict priority for the three highest priority with levels and WFQ for the remaining priority levels.

For additional information on configuring QoS, refer to the chapter entitled "Configuring Quality of Service for the Brocade NetIron XMR and Brocade MLX series".

For examples of queue schedulers for HQoS, refer to HQoS queue scheduler models on page 241.

Supported deployment models

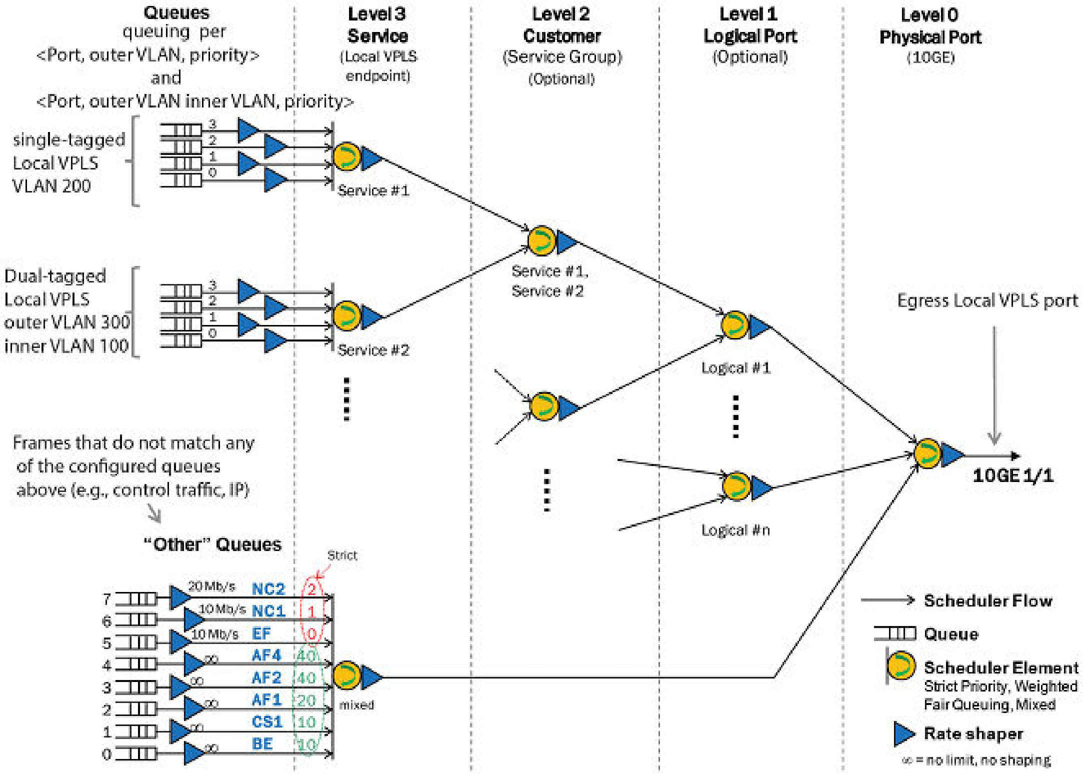
HQoS for Local VPLS

The figure below shows a Local VPLS HQoS model. This model can be used for any kind of VLAN, such as Customer 802.1Q VLANs (CVLAN), Provider Bridging 802.1ad VLANs (SVLAN), or Provider Backbone Bridging 802.1ah VLANs (BVLAN). The type of VLAN being used is defined by the port Ethertype configuration.

As this figure shows, HQoS for Local VPLS supports single-tagged and dual-tagged endpoint queuing on the same egress port.

Traffic that is not explicitly mapped to one of the VLAN queues is sent to the default "other" queues shown in the figure. The parameters of the other queue are configurable as described in Other traffic on page 215.

FIGURE 10 HQoS for Local VPLS with single-tagged and dual-tagged endpoints example

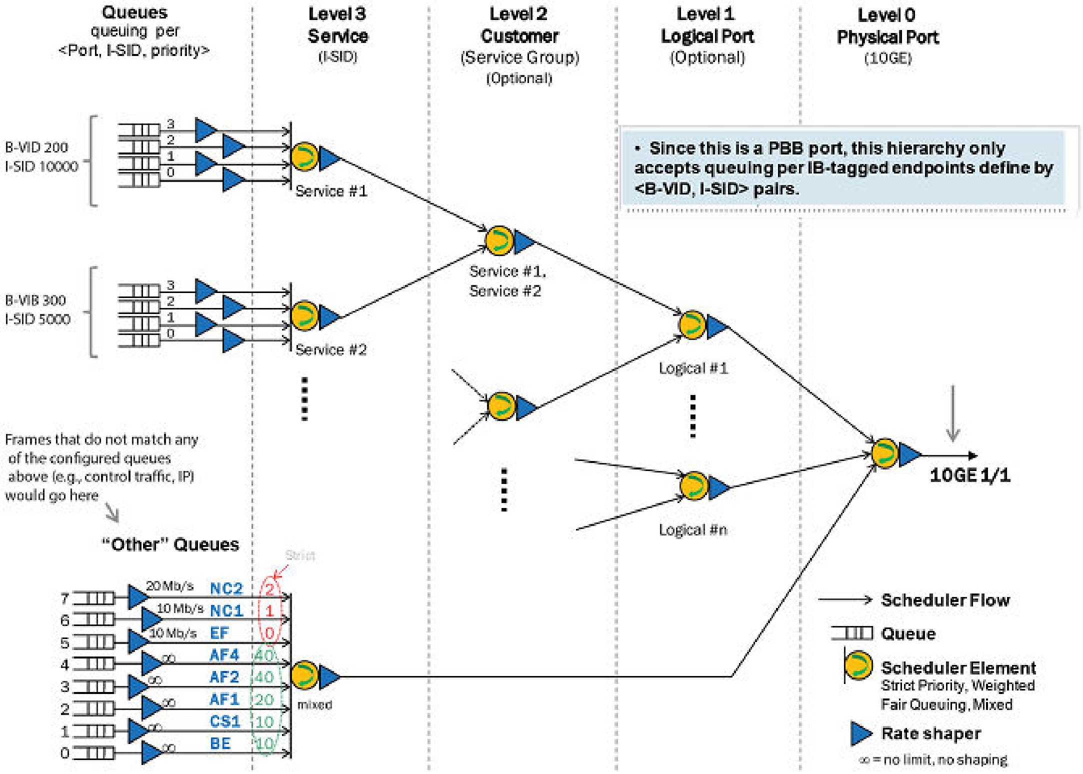


HQoS for PBB traffic

PBB ports can use either BVLAN-based queuing (BVLAN HQoS model, as shown in HQoS for Local VPLS on page 212) or I-SID-based queuing as shown in the figure below, where the egress port is an 802.1ah (PBB) port and where packets are queued per I-SID.

A BVLAN may carry a large number of services identified by distinct I-SID values. The BVLAN HQoS model (using HQoS for Local VPLS as shown in HQoS for Local VPLS on page 212) differs from the I-SID HQoS model in that BVLANs represent PBB tunnels that carry traffic from many different services, while the I-SID HQoS model represent individual services. The B-VLAN HQoS model would is for Backbone Core Bridges, while the I-SID HQoS model is for Backbone Edge Bridges.

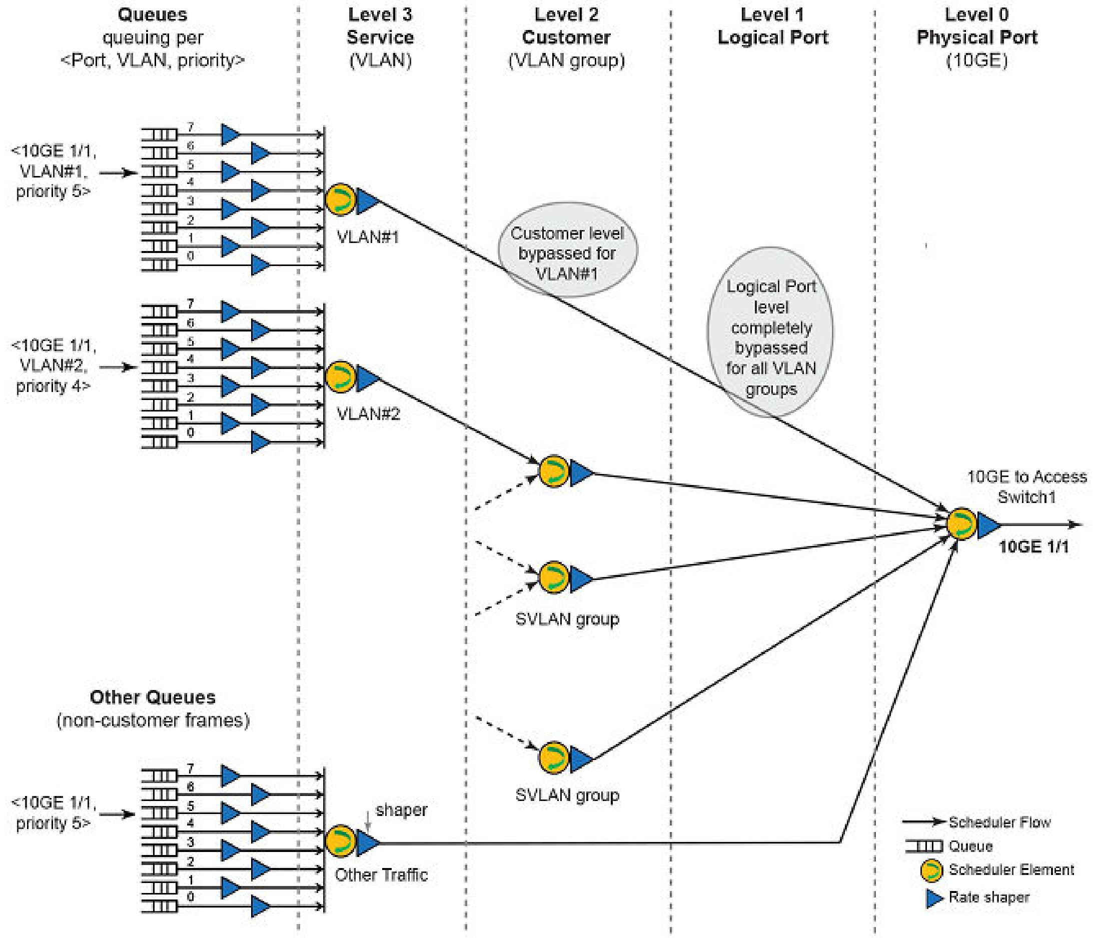
FIGURE 11 PBB HQoS example



Bypassing hierarchy levels

The following figure is an example where the Service Provider does not require the Logical Port level.

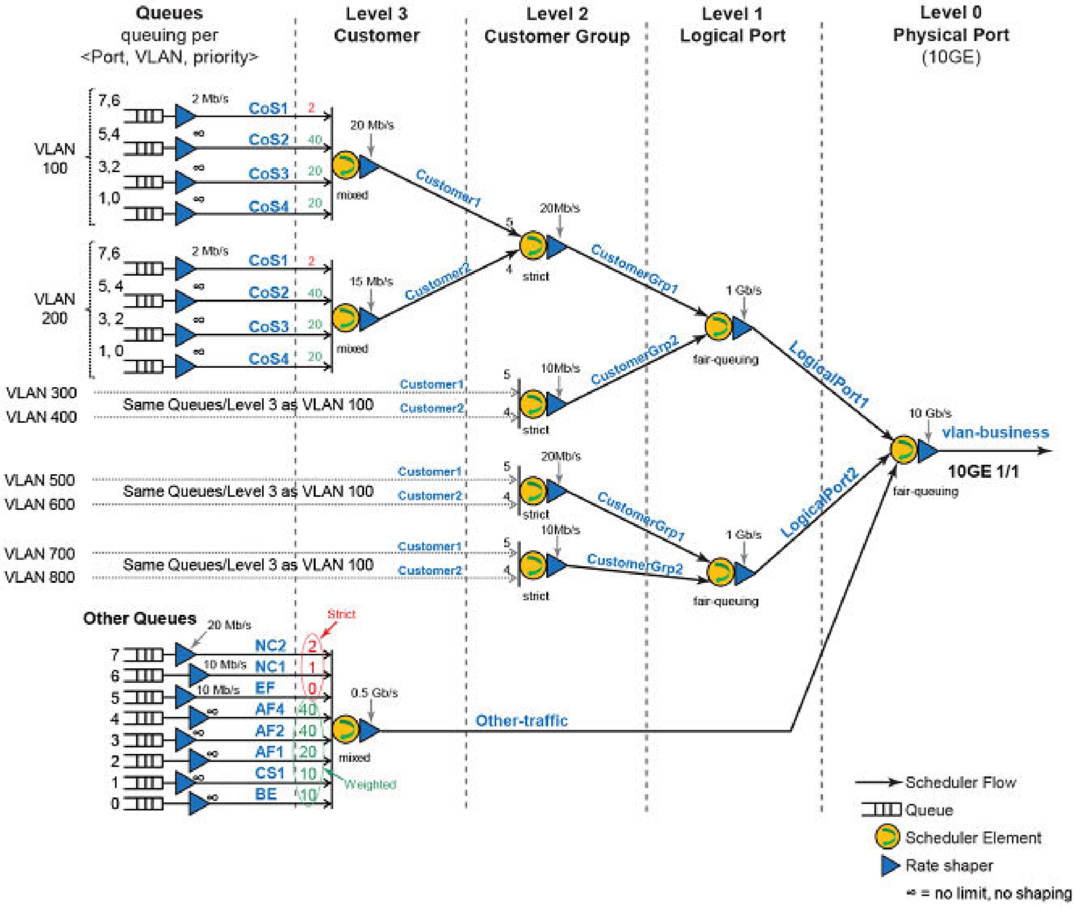
FIGURE 12 Bypassing hierarchy levels example



Other traffic

The following figure displays a Local VPLS HQoS model concurrently supporting non-customer traffic, which is referred to as "other traffic". The customer traffic will always be queued and scheduled through the customer portion of the Layer 2 HQoS scheduler irrespective of what customer Layer 2 traffic is carrying. The "other queues" is used for non-customer traffic. That is, traffic that is not explicitly mapped to an HQoS queue. The "other queues" supports 8 queues/levels of priority. For this kind of traffic, you can queue packets per priority level, as shown. Note: Each priority level can be shaped independently and the other traffic as a whole can also be shaped to a desired rate.

FIGURE 13 Other traffic example

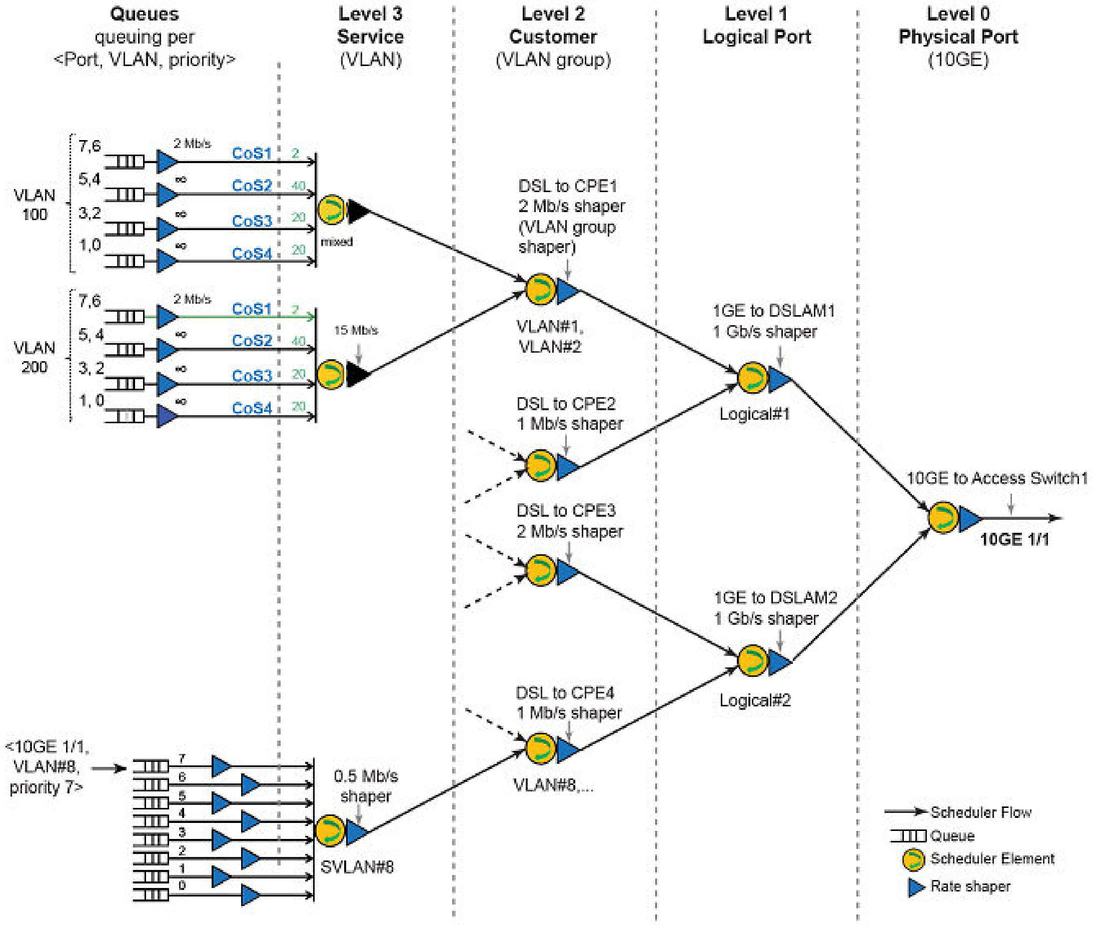


Configuring HQoS

The HQoS configuration procedure goes through the following phases:

- Create scheduling entities and configure forwarding profiles for them.
- Associate the scheduling entities with the forwarding profiles.
- Configure the match criteria for each node.
- Apply the organized scheduler policy to an interface.

FIGURE 14 HQoS example



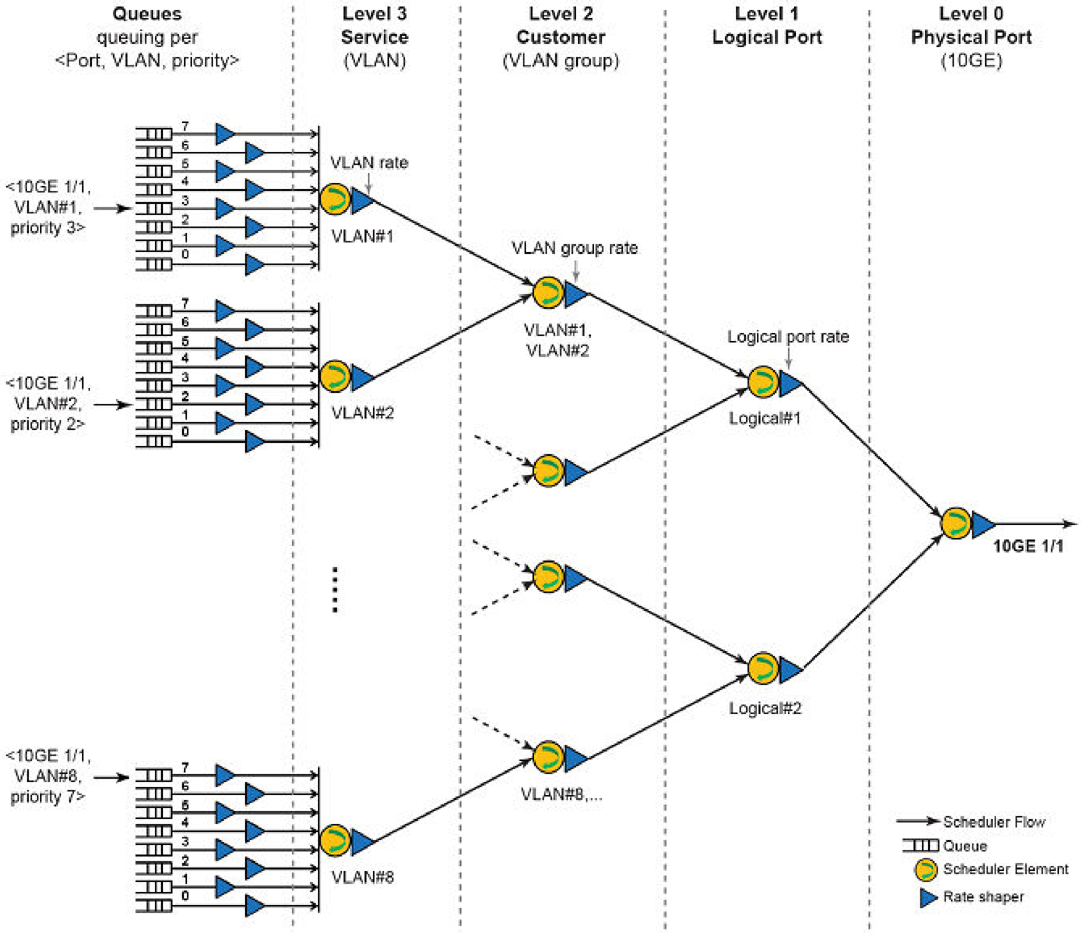
The rate shaper of a scheduler flow will limit the amount of traffic that can be scheduled from that scheduler flow. Therefore, the combination of scheduler flow rate shapers and scheduler element allows for the sharing of bandwidth among the respective scheduler flows.

Configuration considerations

- HQoS is not supported for broadcast, multicast, and unknown unicast traffic
- HQoS is not supported on the interface of a LAG
- If HQoS is enabled on a interface, then the port QoS commands (port shaper configurations) cannot be executed
- Egress mirroring is degraded when HQoS is enabled
- Continuous bursts from 2K HQoS customer streams of jumbo packet sizes at line rate cannot be sustained by XPP's small egress FIFOs.

Configuring HQoS for Local VPLS

FIGURE 15 HQoS for Local VPLS model

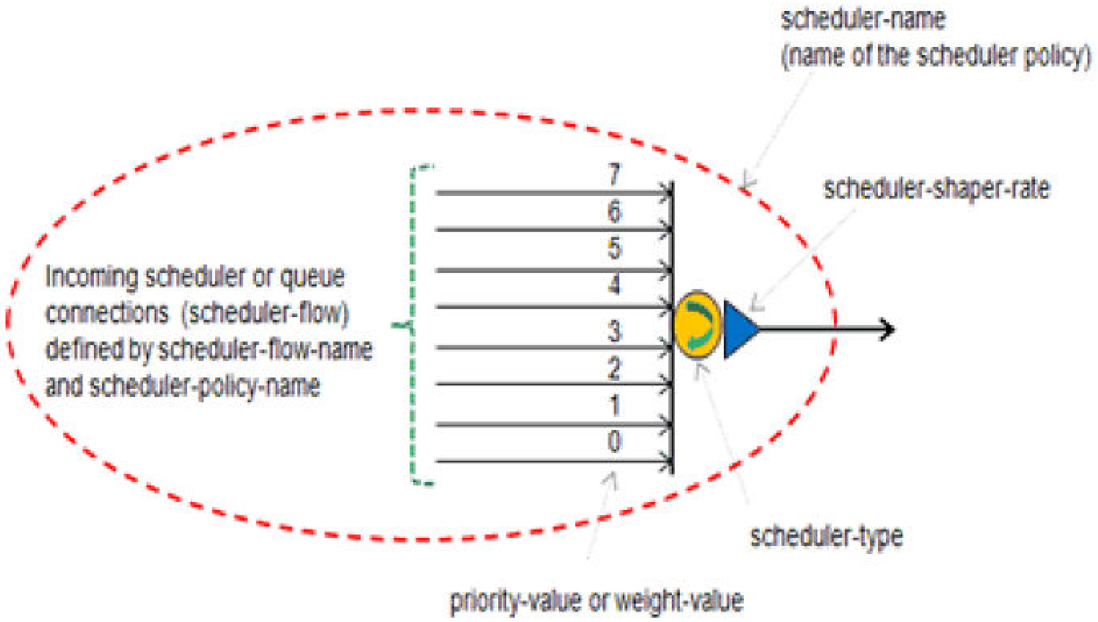


HQoS scheduler policy

HQoS is configured by first defining the HQoS scheduler policies and then mapping them to the physical ports. The same HQoS scheduler policy can be applied to multiple ports to enable provisioning.

To build a complete HQoS scheduler, you must define the types of scheduler elements used by the HQoS hierarchy at each hierarchy level.

FIGURE 16 HQoS scheduler policy


scheduler-name
(name of the scheduler policy)
scheduler-shaper-rate
Incoming scheduler or queue connections (scheduler-flow) defined by scheduler-flow-name and scheduler-policy-name
7
6
5
4
3
2
1
0
scheduler-type
priority-value or weight-value

The following HQoS scheduler policies examples refer to Configuring HQoS for Local VPLS on page 218.

Level 0 policy

The following is an example of how to configure a Level 0 policy

```
device(config)# hqos scheduler-policy vlan-business level level-0
device(config-hqos-scheduler-policy vlan-business) #shaper-rate 10000000
device(config-hqos-scheduler-policy vlan-business) #shaper-burst-size 10
device(config-hqos-scheduler-policy vlan-business) #scheduler-type weighted
device(config-hqos-scheduler-policy vlan-business) #scheduler-flow LogicalPort1
scheduler-input 7 scheduler-policy logical-port-type1
device(config-hqos-scheduler-policy vlan-business) #scheduler-flow LogicalPort2
scheduler-input 6 scheduler-policy logical-port-type1
device(config-hqos-scheduler-policy vlan-business) #scheduler-flow Other-traffic
scheduler-input 5 scheduler-policy other-policy
```

Level 1 policy

The following is an example of how to configure a Level 1 policy.

```
device(config)# hqos scheduler-policy logical-port-type1 level level-1
device(config-hqos-scheduler-policy logical-port-type1)# shaper-rate 1000000
device(config-hqos-scheduler-policy logical-port-type1)# shaper-burst-size 10
device(config-hqos-scheduler-policy logical-port-type1)# scheduler-type weighted
device(config-hqos-scheduler-policy logical-port-type1)# scheduler-flow CustomerGrp1
scheduler-input 3 scheduler-policy customer-group-type1
device(config-hqos-scheduler-policy logical-port-type1)# scheduler-flow CustomerGrp2
scheduler-input 2 scheduler-policy customer-group-type1
```

Level 2 policy

The following is an example of how to configure a Level 2 policy.

```
device(config)# hqos scheduler-policy customer-group-type1 level level-2
device(config-hqos-scheduler-policy customer-group-type1)# shaper-rate  20000
device(config-hqos-scheduler-policy customer-group-type1)# shaper-burst-size 10
device(config-hqos-scheduler-policy customer-group-type1)# scheduler-type strict
device(config-hqos-scheduler-policy customer-group-type1)# scheduler-flow Customer1
scheduler-input 3 scheduler-policy customer-type1
device(config-hqos-scheduler-policy customer-group-type1)#scheduler-flow Customer2
scheduler-input 2 scheduler-policy customer-type1
```

Level 3 policy

The following is an example of how to configure a Level 2 policy.

```
device(config)# hqos scheduler-policy customer-type1 level  level-3
device(config-hqos-scheduler-policy customer-type1)# shaper-rate 20000
device(config-hqos-scheduler-policy customer-type1)# shaper-burst-size 10
device(config-hqos-scheduler-policy customer-type1)# scheduler-type mixed
device(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS1 scheduler-
input 3 scheduler-policy Q-7-6
device(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS2 scheduler-
input 2 weight 40 scheduler-policy Q-5-4
device(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS3 scheduler-
input 1 weight 20 scheduler-policy Q-3-2
device(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS4 scheduler-
input 0 weight 20 scheduler-policy Q-1-0
```

Syntax: [no] hqos scheduler-policy *scheduler-policy-name* **level** {**level-0** | **level-1** | **level-2** | **level-3**}

Syntax: [**no**] [**shaper-rate** *shaper-rate*]

Syntax: [**no**] [**shaper-burst-size** *shaper-burst-size*]

Syntax: [**no**] {**scheduler-type** | **scheduler-type-other**} {**strict** | **weighted** | **mixed**}

Syntax: [**no**] **scheduler-flow** *scheduler-flow-name* {**scheduler-input** *scheduler-input-value* | [**weight** *weight-value*] **scheduler-policy** *scheduler-flow-policy-name*}

The *scheduler-policy-name* variable is a string up to 128 characters.

The *scheduler-flow-name* variable is a string up to 128 characters.

The *level-number* variable is one of the following keywords level-0, level-1, level-1, or level-3.

shaper-rate is an optional parameter. The shaping rate is set with the minimum of 1Mbps and a maximum of 10Gbps. If no shaper-rate specified, the traffic will not be subject to shaping.

shaper-burst-size is an optional parameter. The shaper burst size is set with the minimum of 2 Kbytes and a maximum of 256 Kbytes. The default value for the shaper burst size is set to 10 Kbytes.

scheduler-type is either **strict**, **mixed**, or **weighted**. This scheduler is used for 4 queue customer traffic schedulers. For fair-queuing, use weighted scheduler with all weights being equal.

scheduler-type-other is either **strict**, **mixed**, or **weighted**. This scheduler is only used for 8 queue "other traffic" schedulers. For fair-queuing, use weighted scheduler with all weights being equal.

The *scheduler-input-value* is a number representing the ordering of a flow with respect to a scheduler. The range is 0-7.

The *weight-value* is a number representing the weight of a scheduler flow when a weighted or mixed scheduler is used. The range is 1-64.

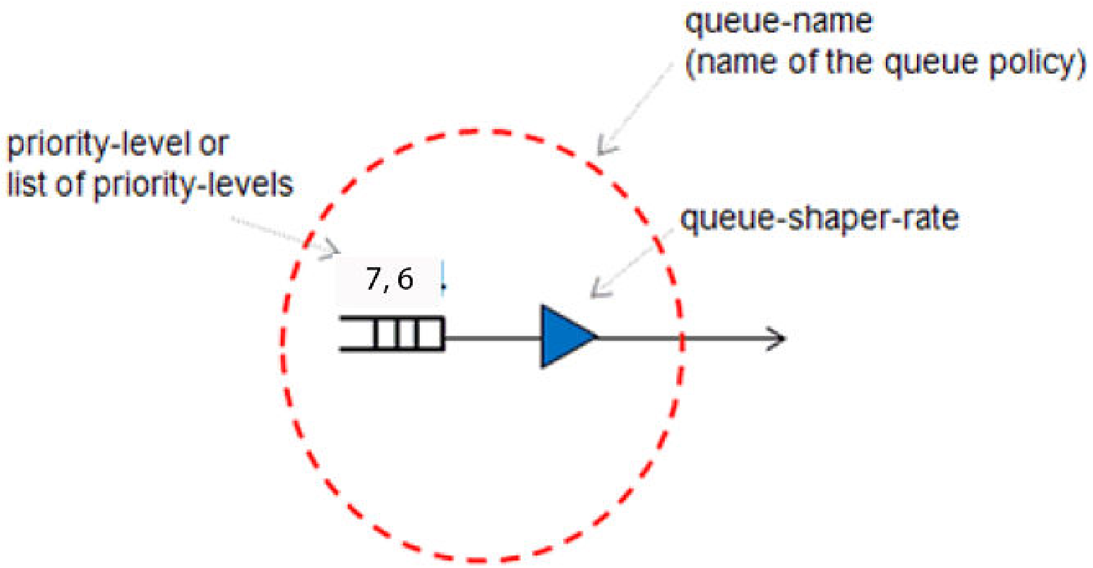
The *scheduler-flow-policy-name* is a string up to 128 characters. The scheduler policy is used for a particular scheduler flow.

HQoS queue policy

HQoS hierarchy is achieved by creating a set of queues.

- A queue is rate shaped and the resulting traffic is referenced as a scheduler flow.
- The scheduler flow out of a queue is referenced by a scheduler policy.
- A queue stores packets based on a selected matching criteria.

FIGURE 17 H-QoS Queue Policy



Configuring HQoS for Local VPLS on page 218 defines queue policies named "Q-7-6", "Q-5-4", "Q-3-2", and "Q-1-0". A queue policy defines a queue for traffic of a given set of priority levels. Traffic out of a queue is shaped to a desired rate.

To configure the queue policies, enter commands such as the following.

```
device(config)# hqos queue-policy Q-7-6
device(config-hqos-queue-policy Q-7-6)# shaper-rate 2000
device(config-hqos-queue-policy Q-7-6)# shaper-burst-size 10
device(config-hqos-queue-policy Q-7-6)# exit

device(config)# hqos queue-policy Q-5-4
device(config-hqos-queue-policy Q-5-4)# shaper-rate 10000000;     default shaper rate
(10G)
device(config-hqos-queue-policy Q-5-4)# shaper-burst-size 10;        default burst
size (10KB)
device(config-hqos-queue-policy Q-5-4)# exit

device(config)# hqos queue-policy Q-3-2
device(config-hqos-queue-policy Q-3-2)# shaper-rate 10000000
device(config-hqos-queue-policy Q-3-2)# shaper-burst-size 10
device(config-hqos-queue-policy Q-3-2)# exit

device(config)# hqos queue-policy Q-1-0
device(config-hqos-queue-policy Q-1-0)# shaper-rate 10000000
device(config-hqos-queue-policy Q-1-0)# shaper-burst-size 10
```

Syntax: [no] hqos queue-policy *queue-name*

Syntax: [**no**] [**shaper-rate** *shaper-rate*]

Syntax: [**no**] [**shaper-burst-size** *shaper-burst-size*]

The *queue-name* is a string up to 128 characters.

shaper-rate is an optional parameter. The shaping rate is set with the minimum of 1Mbps and a maximum of 10Gbps. If no shaper-rate specified, the traffic will not be subject to shaping.

shaper-burst-size is an optional parameter. The shaper burst size is set with the minimum of 2 Kbytes and a maximum of 256 Kbytes. The default value for the shaper burst size is set to 10 Kbytes.

HQoS for LAG traffic overview

This feature allows you to support HQoS for LAG traffic, where the traffic could be to or from the VPLS cloud. The HQoS map is applied on the MPLS uplink. Traffic coming from a VPLS end-point that is part of a LAG, and going out of the MPLS uplink will be processed for HQoS.

Feature highlights

HQoS over LAG is supported for VPLS Endpoint, Local VPLS, and MPLS VPLS Uplink.

- When LAG is undeployed, the HQoS configuration on the primary and all secondary ports will be retained.
- Addition of a new port to the LAG is allowed, if and only if, the HQoS configuration of the newly added port is identical to that of the primary port of the deployed LAG.
- Removal of a port from the deployed LAG with HQoS configuration will retain the HQoS configuration on the port which is being removed from the LAG.
- Before the HQoS configuration is applied on the primary port of a deployed LAG, and the configuration is replicated on all the secondary ports of the LAG, the following checks are made.
  1. It is ensured that the resources are available (per TM).
  2. All member ports are 8x10G ports that support HQoS when the policy is applied. Different kinds of 10G ports are not mixed.
  3. If the member port list contains ports that are not HQoS capable, the CLI command flags an error and disallows the command execution.
  4. When unbinding an HQoS policy from a port, HQoS policy is removed from all member ports and resources are de-allocated from all the member ports.

Configuration procedure

HQoS over LAG is configured under the primary port of the LAG.

1. Use this command to set up a LAG and Primary port.

```
device # lag “testLag” dynamic id 1
ports ethernet 4/3 to 4/5
primary-port 4/3
deploy
```

2. Use the following commands to configure HQoS on the primary port of the LAG.

```
device (config)# interface ethernet 4/3
device (config-if-eth-4/3) # hqos service-policy output policy-1
device (config-if-eth-4/3) # hqos-map flow-1-1.flow-2-1.flow-3-1 match vlan 200
(Existing VPLS End-point)
device (config-if-eth-4/3) # hqos-map flow-1-1.flow-2-1.flow-3-0 match vpls 501
peer 1.1.1.2 (VPLS MPLS Uplink)
device (config-if-eth-4/3)# enable
```

The HQoS configuration will be replicated on both the secondary ports (4/4, 4/5) of the LAG.Depending on the traffic patterns and the hash function used, lag hashing may result in non-uniform distribution of traffic to member ports. Each member port is individually capable of forwarding the traffic which is configured as part of the corresponding HQoS-policy and HQoS-map rule. The HQoS over LAG is supported both for the VPLS End-point & VPLS MPLS Uplink.

Limitations

All member ports need to have the same HQoS configuration before the LAG can be deployed. This condition covers the following cases.

- No HQoS configuration exists on any member ports.
- HQoS configuration on all member ports is the same.
- If no HQoS configuration exists on any member ports, member ports with different capabilities (HQoS capable and non-HQoS capable) will be allowed.

HQoS for VPLS traffic overview

This feature allows you to support HQoS for VPLS traffic, where the traffic could be to or from the VPLS cloud. The HQoS map is applied on the MPLS uplink. Traffic coming from a VPLS end-point and going out of the MPLS uplink will be processed for HQoS.

Feature highlights

HQoS was previously supported for "local VPLS" only. This feature is an enhancement to allow HQoS for VPLS in addition to local VPLS. A new match condition containing the VPLS ID and the VPLS Peer IP address has been added to the HQoS map command.

Configuring HQoS for VPLS traffic

These steps assume the following topology:

- PE11 and PE12 routers are MCT nodes
- PE3 (1.1.1.2) is the remote PE router
- PE11 and PE12 are connected through MPLS

Use the following commands to configure HQoS policy on Node PE11.

```
device (config)# HQOS scheduler-policy policy-1 level level-0
device (config-hqos-scheduler-policy policy-1)# shaper-rate 1000000
device (config-hqos-scheduler-policy policy-1)# shaper-burst-size 128
device (config-hqos-scheduler-policy policy-1)# scheduler-type strict
device (config-hqos-scheduler-policy policy-1)# scheduler-flow flow-1-0
scheduler- input 0 scheduler-policy policy-2
device (config-hqos-scheduler-policy policy-1)# scheduler-flow flow-1-1
device input 1 scheduler-policy policy-2
device (config-hqos-scheduler-policy policy-1)#!
device (config-hqos-scheduler-policy policy-1)#HQOS scheduler-policy policy-2 leve l
level-1
device (config-hqos-scheduler-policy policy-2)# shaper-rate 1000000
device (config-hqos-scheduler-policy policy-2)# shaper-burst-size 64
device (config-hqos-scheduler-policy policy-2)# scheduler-type strict
device (config-hqos-scheduler-policy policy-2)# scheduler-flow flow-2-0
scheduler- input 0 scheduler-policy policy-3
device (config-hqos-scheduler-policy policy-2)# scheduler-flow flow-2-1
scheduler- input 1 scheduler-policy policy-3
device (config-hqos-scheduler-policy policy-2)#
device (config-hqos-scheduler-policy policy-2)#HQOS scheduler-policy policy-3
leve l level-2
device (config-hqos-scheduler-policy policy-3)# shaper-rate 20000
device (config-hqos-scheduler-policy policy-3)# shaper-burst-size 64
device (config-hqos-scheduler-policy policy-3)# scheduler-type strict
device (config-hqos-scheduler-policy policy-3)# scheduler-flow flow-3-0
scheduler- input 0 scheduler-policy policy-4
device (config-hqos-scheduler-policy policy-3)# scheduler-flow flow-3-1
scheduler- input 1 scheduler-policy policy-4
device (config-hqos-scheduler-policy policy-3)#!
device (config-hqos-scheduler-policy policy-3)#HQOS scheduler-policy policy-4
leve l level-3
device (config-hqos-scheduler-policy policy-4)# shaper-rate 2000
device (config-hqos-scheduler-policy policy-4)# shaper-burst-size 10
device (config-hqos-scheduler-policy policy-4)# scheduler-type strict
device (config)#router mpls
device (config-mpls)#mpls-interface e3/3
device (config-mpls-if-e100-3/3)#mpls-interface ve 200
device (config-mpls-if-ve-200)#
device (config-mpls-if-ve-200)# vpls test1 5000
device (config-mpls-vpls-test1)# vpls-peer 1.1.1.2
device (config-mpls-vpls-test1)# vlan 100
device (config-mpls-vpls-test1-vlan-100)# tagged ethe 4/1
```

Use the following commands to configure HQoS policy on Node PE11.

```
device (config)# interface ethernet 3/3
device (config-if-eth-3/3) # hqos service-policy output policy-1
device (config-if-eth-3/3) # hqos-map flow-1-1.flow-2-1.flow-3-1 match vpls 5000
peer 1.1.1.2
device (config-if-eth-3/3)# enable
```

Limitations

- The same configuration must be applied on both MCT nodes.
- Any module (except BR-MLX-10Gx24 and BR-MLX-40Gx4-X) can be used for ingress traffic destined for an HQoS port. Only BR-MLX-10Gx8-M and BR-MLX-10Gx8-X modules support egressing HQoS traffic.
- It is recommended that you configure the HQoS Map on all the MPLS Uplink interfaces.
- BGP Auto-discovery for VPLS is not supported.
- HQoS will not work properly in MCT VPLS failure scenario e.g. CCP-DOWN or Spoke Down.

Checking for HQoS for VPLS configurations on ports

Example 1:

```
device #show run int e 3/3
interface ethernet 3/3
hqos service-policy output policy-1
hqos-map flow-1-1.flow-2-1.flow-3-1 match vpls 5000 peer 1.1.1.2
enable
```

After a successful configuration on PE11, the **show run interface** command output indicates that HQoS will be applied to traffic coming from VPLS 5000 endpoint and going to VPLS peer 1.1.1.2 on the MPLS interface eth 3/3.

Example 2:

```
device #show run int e 4/1
interface ethernet 4/1
hqos service-policy output policy-1
hqos-map flow-1-1.flow-2-1.flow-3-1 match vlan 100
enable
```

After a successful configuration on PE11, the **show run interface** command output indicates that HQoS will be applied to traffic coming from peer 1.1.1.2 and going to VPLS 5000 endpoint, interface eth 4/1.

WRED support for HQoS

This feature allows you to support WRED for HQoS customer and other queue types.

Feature highlights

This feature is implemented using enhancements to existing CLI commands for QoS on regular port queue types.

Configuring steps

1. Use the following commands to set up WRED on a 10G module and a customer-queue type.

```
device # hqos customer-queue-type 0 wred enable module-type 8x10g
device # hqos customer-queue-type 0 wred averaging-weight 1 module-type 8x10g
```

2. Use the following commands to set up WRED on a 10G module and an other-queue type.

```
device # hqos other-queue-type 7 wred enable module-type 8x10g
device # hqos other-queue-type 7 wred drop-precedence 3 max-avg-queue-size 512
module-type 8x10g
```

Use the show command to check your configuration. After a successful configuration, the show command output will be similar to what is shown in the example below.

```
device #show hqos wred module-type 8x10g
Other Traffic
QType  Enable  AverWeight  MaxQSz  DropPrec  MinAvgQSz  MaxAvgQSz  MaxDropProb
MaxPktSz
 0      No
 1    No
 2     No
 3      No
 4     No
 5     No
 6     No
 7     Yes     4(6.25%)  1024        0          1024       1088      0%        16384
                      1        704        832           2%     16384
                      2        448        832        5%     16384
                      3        384        512        6%     16384
Customer Traffic
 0     Yes     1(50.0%)  1024         0         384        1024      2%     16384
                     1        320         1024       4%     16384
                     2        256        1024       9%     16384
                     3        192        1024       10%    16384
 1     No
 2     No
 3     No
```

WRED Commands

Syntax: [no] hqos { **customer-queue-type** | **other-queue-type** } *queue-type* [**wred enable module-type** *module-type*]

Syntax: [no] hqos { **customer-queue-type** | **other-queue-type** } *queue-type* [**wred averaging-weight** *avg-weight-value* **module-type***module-type*]

Syntax: [no] hqos { **customer-queue-type** | **other-queue-type** } *queue-type* [**wred drop-precedence** *drop-precedence-value* **max-avg-queue-size** | **min-avg-queue-size** *min-size* | *max-size***module-type***module-type*]

Syntax: [no] hqos { **customer-queue-type** | **other-queue-type** } *queue-type* [**wred drop-precedence** *drop-precedence-value* **drop-probability-max** *p-max* **module-type***module-type*]

Syntax: [no] hqos { **customer-queue-type** | **other-queue-type** } *queue-type* [**wred drop-precedence** *drop-precedence-value* **packet-size-max** *pkt-size* **module-type***module-type*]

Syntax: [no] hqos { **customer-queue-type** | **other-queue-type** } *queue-type* [**wred drop-precedence** *drop-precedence-value* **default-params module-type***module-type*]

Syntax: [no] hqos { **customer-queue-type** | **other-queue-type** } *queue-type* [**default-params module-type***module-type*]

Syntax: show hqos [**wred module-type** *module-type*]

Binding a policy to an interface and applying a mapping condition

The **hqos service-policy** command is used to apply an HQoS scheduler policy to an egress physical interface or port. An 8 input Strict Priority (SP) scheduler will be created on the egress port if no other match condition is specified through the **hqos-map** command.

NOTE
This command will be valid only for interfaces on NI-MLX-10Gx8-M and NI-MLX-10Gx8-X modules. An error will be seen for interfaces on the NI-MLX-10Gx8-D module.

The **hqos-map** command is used to map customer endpoints and profiles to the HQoS scheduler flows. The mapping of customer endpoints and profiles is specified through the match criteria. The **hqos- map** command allows for changing the shaping rate of a scheduler. When configuring the interface, consider the following:

- Within the port configuration, **hqos-map** keyword must be present after **service-policy output** *level-0-scheduler-name*
- Only level 0 policies can be applied to an interface.
- If an HQoS policy is applied to an interface, the following changes to the scheduling tree structure is not supported.
  - Modifying HQoS policies (scheduler or queue)
  - Adding or removing a scheduler or queue policy that is referenced in the bound policy

The following is an example of how to configure the HQoS policy mapping for the VLAN HQoS model defined in Configuring HQoS for Local VPLS on page 218.

Map the vlan-business policy to the Ethernet port 1/1 by using commands such as the following.

```
device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# hqos service-policy output vlan-business
```

Syntax: interface ethernet *port/slot*

Syntax: [no] hqos service-policy output *level-0-scheduler-name*

The **service-policy output** option defines the *level-0-scheduler-name* and can be up to 128 characters.

Make the necessary VLAN mappings and settings to the shaper rates by using commands such as the following.

```
device(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp1.Customer1 match
vlan 100
device(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp1.Customer2 match
vlan 200
device(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp2.Customer1 match
vlan 300
device(config-if-e10000-1/1)#hqos-map LogicalPort1.CustomerGrp2.Customer2 match vlan
400
device(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp1.Customer1 match
vlan 500
device(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp1.Customer2 match
vlan 600
device(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp2.Customer1 match
vlan 700
device(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp2.Customer2 match
vlan 800
device(config-if-e10000-1/1)# hqos-map Other-traffic match other
device(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp1.Customer2 shaper-
rate 15000
device(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp2 shaper-rate 10000
device(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp2 shaper-rate 10000
```

Syntax: [no] hqos-map *hqos-scheduler-node-name*

Syntax: [**no**] [shaper-rate *shaper-rate*]

Syntax: [**no**] [**shaper-burst-size** *shaper-burst-size*]

Syntax: [**no**] {**match other** | **match vlan** *vlan-num*} | {**inner-vlan** *inner-vlan-num* | **isid** *isid-num*}

The *hqos-scheduler-node-name* is a string up to 512 characters. The scheduler-node-name specifies the hqos-scheduler-node in full-path format.

The **shaper-rate** is an optional parameter. If there is no shaper-rate specified, the shaper-rate as specified in the HQoS policy will be used.

The **inner-vlan** is an optional parameter. The *inner-vlan-num* range is 1 through 4094. This option should be the egress VLAN on the HQoS port.

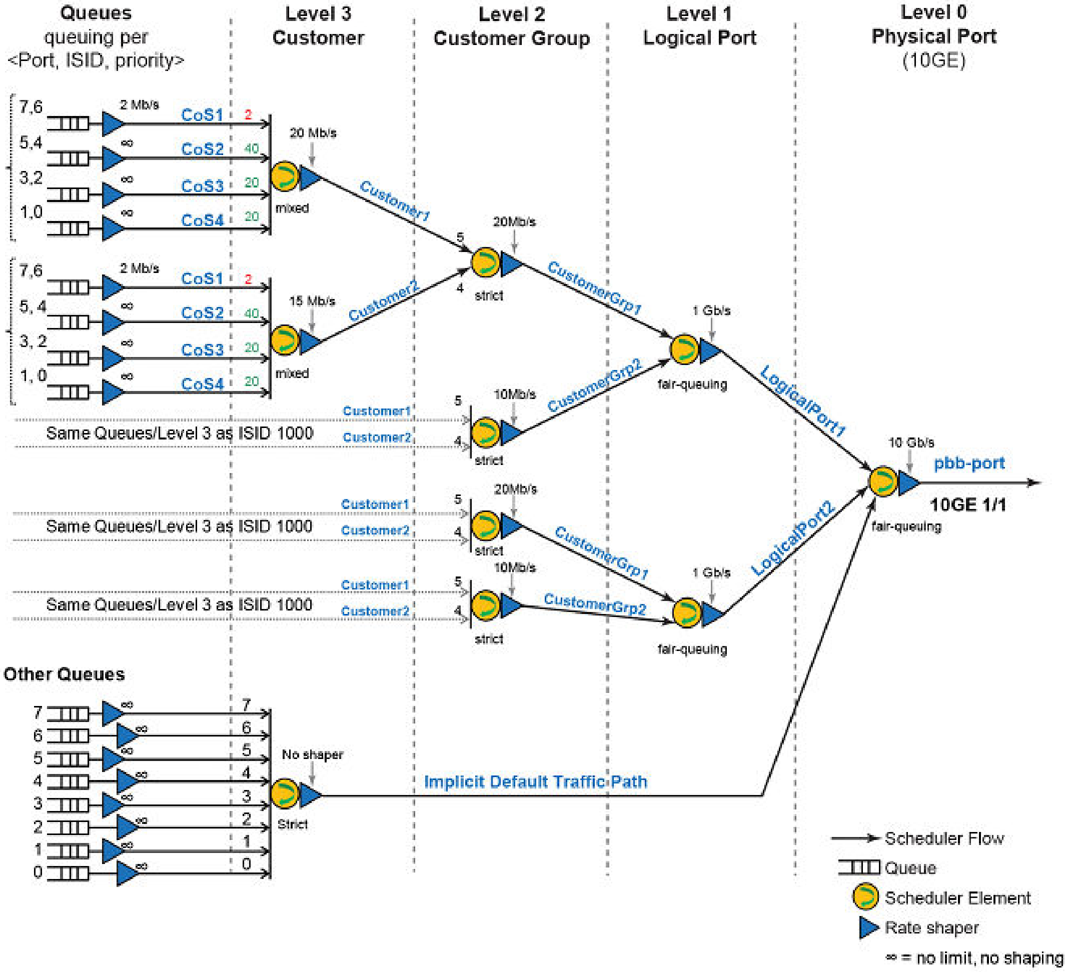
The **shaper-burst-size** is an optional parameter. If there is no shaper-burst-size specified, the shaper-burst-size as specified in the HQoS policy will be used.

The **match other** is an optional parameter. For traffic that does not match any of the other defined matching rules. This is the default matching criteria for a scheduler flow without any defined matching criteria.

The **match vlan** is an optional parameter. The *vlan-num* range is 1 through 4094. This option should be the egress VLAN on the HQoS port.

PBB HQoS

FIGURE 18 PBB HQoS mode



HQoS configuration for PBB

Refer to HQoS scheduler policy on page 218 for additional information on configuring the HQoS scheduler policies.

At the level-0 scheduler policy configuration, the main difference is the absence of an explicit "default traffic" path. A default path is created implicitly with strict priority scheduling among the 8 default traffic queues. The implicit default traffic path is always lower priority than customer traffic regardless of level-0 scheduler type.

```
device(config)# hqos scheduler-policy pbb-port level level-0
device(config-hqos-scheduler-policy pbb-port)# shaper-rate 10000000
device(config-hqos-scheduler-policy pbb-port)# shaper-burst-size 10
device(config-hqos-scheduler-policy pbb-port)# scheduler-type weighted
device(config-hqos-scheduler-policy pbb-port)# scheduler-flow LogicalPort1 scheduler-
input 7 scheduler-policy logical-port-type1
device(config-hqos-scheduler-policy pbb-port)# scheduler-flow LogicalPort2 scheduler-
input 6 scheduler-policy logical-port-type1
```

At the level-1 scheduler policy configuration, there are two customer groups competing in a weighted fair queue.

- CustomerGrp1 will get preferential treatment with twice the bandwidth of CustomerGrp2 because of the weight values associated with them.
- CustomerGrp1 can receive up to 666 Mbps and CustomerGrp2 receives up to 333 Mbps from the total 1Gbps shaped at this level.

```
device(config)# hqos scheduler-policy logical-port-type1 level level-1
device(config-hqos-scheduler-logical-port-type1)# shaper-rate 1000000
device(config-hqos-scheduler-logical-port-type1)# shaper-burst-size 10
device(config-hqos-scheduler-logical-port-type1)# scheduler-type weighted
device(config-hqos-scheduler-logical-port-type1)# scheduler-flow CustomerGrp1
scheduler-input 3 weight 2 scheduler-policy customer-group-type1
device(config-hqos-scheduler-logical-port-type1)# scheduler-flow CustomerGrp2
scheduler-input 2 weight 1 scheduler-policy customer-group-type1
```

At the level-2 scheduler policy configuration, the only thing that changes is the scheduling type from strict to fair weighted (when no explicit weight value is set, the default is 1).

```
device(config)# hqos scheduler-policy customer-group-type1 level level-2
device(config-hqos-scheduler-customer-group-type1)# shaper-rate  20000
device(config-hqos-scheduler-customer-group-type1)# shaper-burst-size 10
device(config-hqos-scheduler-customer-group-type1)# scheduler-type weighted
device(config-hqos-scheduler-customer-group-type1)# scheduler-flow Customer1
scheduler-input 3 scheduler-policy customer-type1
device(config-hqos-scheduler-customer-group-type1)# scheduler-flow Customer2
scheduler-input 2 scheduler-policy customer-type1
```

At the last level, each customer employs a strict scheduler amongst its 4 priority queues with no shapers (open shapers set to 10Gbps).

```
device(config)# hqos scheduler-policy customer-type1 level  level-3
device(config-hqos-scheduler-customer-type1)# shaper-rate 20000
device(config-hqos-scheduler-customer-type1)# shaper-burst-size 10
device(config-hqos-scheduler-customer-type1)# scheduler-type strict
device(config-hqos-scheduler-customer-type1)# scheduler-flow CoS1 scheduler-input 3
scheduler-policy Q-7-6
device(config-hqos-scheduler-customer-type1)# scheduler-flow CoS2 scheduler-input 2
scheduler-policy Q-5-4
device(config-hqos-scheduler-customer-type1)# scheduler-flow CoS3 scheduler-input 1
scheduler-policy Q-3-2
device(config-hqos-scheduler-customer-type1)# scheduler-flow CoS4 scheduler-input 0
scheduler-policy Q-1-0

device(config)# hqos queue-policy queue-default
device(config)# shaper-rate 10000000
device(config)# shaper-burst-size 10
```

Binding Policy to interface and applying mapping condition for PBB

Refer to Binding a policy to an interface and applying a mapping condition on page 226 for additional information on binding a policy to an interface and applying a mapping condition.

Binding the mappings to HQoS PBB port is similar to the Local VPLS HQoS model. The only difference is you must add the "ISID" as the customer match condition.

```
device(config)# interface ethernet 1/1
device(config-if-e10000-1/1)# hqos service-policy output pbb-port
device(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp1.Customer1 match
vlan 100 isid 1000
device(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp1.Customer2 match
vlan 200 isid 2000
device(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp2.Customer1 match
vlan 300 isid 3000
device(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp2.Customer2 match
vlan 400 isid 4000
device(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp1.Customer1 match
vlan 500 isid 5000
```

```
device(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp1.Customer2 match
vlan 600 isid 6000
device(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp2.Customer1 match
vlan 700 isid 7000
device(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp2.Customer2 match
vlan 800 isid 8000
```

Displaying HQoS information

Use the following commands to show HQoS information.

Displaying the HQoS policy

Use the **show hqos policy** command to display the information of the policy and its associated flows.

```
device# show hqos policy vlan-business
    Scheduler-policy-name vlan-business    Level level-0
    Scheduler-type weighted          Shaper-rate 2000000 Kbps
    scheduler-flow Logical-Port1 scheduler-input 7 scheduler-policy logical-port-
type1
    scheduler-flow Logical-Port2 scheduler-input 6 scheduler-policy logical-port-
type1
    scheduler-flow Other-traffic  scheduler-input 5 scheduler-policy other-policy
```

Use the **show hqos policy** command to display the information of all the interfaces to which this policy is applied.

```
device# show hqos policy vlan-business applied
            Ethernet Port:  1/1
            Scheduler-node Type:  Root
            Scheduler-node Policy Name:  vlan-business
            Scheduler-node Name:  vlan-business
            Scheduler-node ID:  0x00610000
```

Syntax: show hqos policy *policy-name* [**applied**]

Displaying HQoS interface information

Use the **show hqos interface ethernet** command to display information for the specified interface.

```
device# show hqos interface eth 1/1
    Interface Number:         0x31
    HQOS State:         Enabled
    Scheduler Tree State:     Download Finish/Active
    Policy-name:         vlan-business

    Scheduler-Node Type:             Root
    Scheduler-Node Name:             vlan-business
    Scheduler-Node ID:             0x310000
    Scheduler-Node Scheduler Type:         Strict
    Scheduler-Node Shaper Rate:         2000000 Kbps
    Scheduler-Node Shaper Rate Burst Size:     128 KB
```

In this example, the HQoS policy tree has been applied to an interface.

```
device# show hqos interface eth 1/1 scheduler-node vlan-business.Logical-Port1
    Scheduler-Node Type:             Non-Root
    Scheduler-Node Policy Name:                                     logical-port-
type1
    Scheduler-Node Name:             Logical-Port1
    Scheduler-Node ID:             0x310008
    Scheduler-Node Scheduler Type:         Strict
    Scheduler-Node Scheduler-input:         1
    Scheduler-Node Shaper Rate:         1000000 Kbps
    Scheduler-Node Shaper Rate Burst Size:     64 KB
    Scheduler-Node-Parent Node Name:         vlan-business
    Scheduler-Node-Parent Node ID:         0x310000
```

In this example, the HQoS policy tree has been applied to an interface and shows the parent and child node information in a HQoS policy tree.

```
device# show hqos interface eth 1/1 scheduler-node vlan-business.Logical-Port1 child
    Parent Information:
    Scheduler-Node Type:             Non-Root
    Scheduler-Node Policy Name: logical-port-type1
    Scheduler-Node Name:             Logical-Port1
    Scheduler-Node ID:             0x310008
    Scheduler-Node Scheduler Type:         Strict
    Scheduler-Node Scheduler-input:         1
    Scheduler-Node Shaper Rate:         1000000 Kbps
    Scheduler-Node Shaper Rate Burst Size:     64 KB
    Scheduler-Node-Parent Node Name:         vlan-business
    Scheduler-Node-Parent Node ID:         0x310000

    Child Information:
    Scheduler-Node Type:             Non-Root
    Scheduler-Node Policy Name: customer-type1
    Scheduler-Node Name:             Logical-Port1. Customer1
Scheduler-Node ID:                 0x310015
    Scheduler-Node Scheduler Type:             Strict
    Scheduler-Node Scheduler-input:             3
    Scheduler-Node Shaper Rate:             20000 Kbps
    Scheduler-Node Shaper Rate Burst Size:         128 KB
    Scheduler-Node-Parent Node Name:             Logical-Port1
    Scheduler-Node-Parent Node ID:             0x310008

    Scheduler-Node Type:                 Non-Root
    Scheduler-Node Policy Name: customer-type1
    Scheduler-Node Name:                 Logical-Port1.Customer2
    Scheduler-Node ID:                 0x310016
    Scheduler-Node Scheduler Type:             Strict
    Scheduler-Node Scheduler-input:             3
    Scheduler-Node Shaper Rate:             20000 Kbps
    Scheduler-Node Shaper Rate Burst Size:         128 KB
    Scheduler-Node-Parent Node Name:             Logical-Port1
    Scheduler-Node-Parent Node ID:             0x310008
```

Syntax: show hqos interface ethernet *slot/port* **scheduler-node** {*scheduler-node-name* | *scheduler-node-id*} [**child**]

Displaying the HQoS Max Queue Size

Use the **show hqos max-queue-size** command to display the priority for customer and default traffic queues.

```
device# show hqos max-queue-size
Other Traffic
QType Max-Size (KB)
-----+-------------
  0       1024
  1       1024
  2       1024
  3       1024
  4       1024
  5       1024
  6       1024
  7       1024
Customer Traffic
QType Max-Size (KB)
-----+-------------
  0       1024
  1       1024
  2       1024
  3       1024
```

Syntax: show hqos max-queue-size

Displaying the buffer pool

Use the **show hqos buffer-pool** command to display the HQoS buffer pool configurations.

```
device# show hqos buffer-pool

Other Traffic
QType Buffer Type
-----+-------------
  0   BRONZE
  1   BRONZE
  2   BRONZE
  3   BRONZE
  4   BRONZE
  5   BRONZE
  6   BRONZE
  7   GOLD

Customer Traffic
QType Buffer Type
-----+-------------
  0   BRONZE
  1   BRONZE
  2   BRONZE
  3   GOLD

Buffer Type    Memory(%)    Min. Gurantee(%)

BRONZE                95    0
GOLD                 100    5

Module Type  Total Memory  Max. Gold  Min. Gold  Max. Bronze  Min. Bronze
8x10              1392 MB    1392 MB      69 MB      1322 MB         0 MB
```

Syntax: show hqos buffer-pool

Displaying HQoS global resource information

Use the **show hqos resource global** command to display the HQoS resources for specified slot.

```
device# show hqos resource global
Global Resource:
                       Maximum         Allocated
Scheduler:          262144          122
    Queue:          131072          8
show the hqos resources for system
show hqos resource slot 1
Port 2/1 - 2/4:
                    Maximum         Allocated
Scheduler:          7936            122
    Queue:          8192            8
Port 2/5 - 2/8:
                    Maximum         Allocated
Scheduler:          7936            0
    Queue:          8192            0
```

Syntax: show hqos resource global

Displaying HQoS errors

Use the **show hqos error** command to display the HQoS error counts for a specified slot.

```
device# show hqos error slot 2
Slot 2
-------
      Invalid Input Error Count: 0
    No HW Resources Error Count: 0
      Memory Alloc Failed Count: 0
           Internal Error Count: 0
```

```
            LP Busy Error Count: 0
          HW Driver Error Count: 0
            Unknown Error Count: 0
```

Syntax:show hqos error

Displaying HQoS Statistics

Use the **show hqos statistics** command to display the specified flow information for a specified interface.

```
device# show hqos statistics ethernet 1/1 queue Queue name: Queue name: vlan-
business.Logical-Port1.Customer1.COS1
    Priorities: 7, 6
    EnQue Pkt Count 0
    EnQue Bytes Count 0
    DeQue Pkt Count 0
    DeQue Bytes Count 0
    Total Discard Pkt Count 0
Total Discard Bytes Count 0
Oldest Discard Pkt Count 0
Oldest Discard Bytes Count     0
Current Queue Depth  0
Maximum Queue Depth since Last read     0
```

Use the **show hqos statistics** command to display flow information for the "other" traffic flows.

```
device# show hqos statistics ethernet 1/1 queue default-other

Node: implicit_match_all        Queue index: 0
Priorities: 0
                            EnQueue Packet Count                  0
                                EnQueue Byte Count                0
                            DeQueue Packet Count                  0
                                DeQueue Byte Count                0
                      Total Discard Packet Count                  0
                          Total Discard Byte Count                0
                    Oldest Discard Packet Count                   0
                        Oldest Discard Byte Count                 0
                                Current Queue Depth               0
Maximum Queue Depth Since Last Read                               0
```

Use the **show hqos statistics** command to display flow information for the specified index.

```
device#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue
LogicalPort1.CustomerGrp1.Customer1
Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 0
Priorities: 1,0
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 1
Priorities: 3,2
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 2
```

```
Priorities: 5,4
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 3
Priorities: 7,6
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0
```

Use the **show hqos statistics** command to display flow information for the default other flows.

```
device#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue default-
other
Node: implicit_match_all        Queue index: 0
Priorities: 0
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: implicit_match_all        Queue index: 1
Priorities: 1
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: implicit_match_all        Queue index: 2
Priorities: 2
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: implicit_match_all        Queue index: 3
Priorities: 3
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
```

```
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: implicit_match_all        Queue index: 4
Priorities: 4
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: implicit_match_all        Queue index: 5
Priorities: 5
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: implicit_match_all        Queue index: 6
Priorities: 6
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0

Node: implicit_match_all        Queue index: 7
Priorities: 7
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0
```

Use the **show hqos statistics** command to display flow information for the default other flows for an index.

```
device#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue default-other
index 0
Queue name: implicit_match_all
Priorities: 0
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0
```

Use the **show hqos statistics** command to display flow information for the default other flows for a specific index. The following examples displays index 1.

```
device#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue default-other
index 1
```

```
Queue name: implicit_match_all
Priorities: 1
               EnQueue Packet Count                     0
                 EnQueue Byte Count                     0
               DeQueue Packet Count                     0
                 DeQueue Byte Count                     0
         Total Discard Packet Count                     0
           Total Discard Byte Count                     0
        Oldest Discard Packet Count                     0
          Oldest Discard Byte Count                     0
                Current Queue Depth                     0
Maximum Queue Depth Since Last Read                     0
```

Syntax: show hqos statistics ethernet *slot/port* **queue** *hqos-scheduler-node-name* [**index** *index-number*]

Clearing HQoS statistics

Use the **clear the hqos statistics** command to clear the statistics for a specified flow.

```
device# clear hqos statistics ethernet ethernet 1/1 queue default-other
```

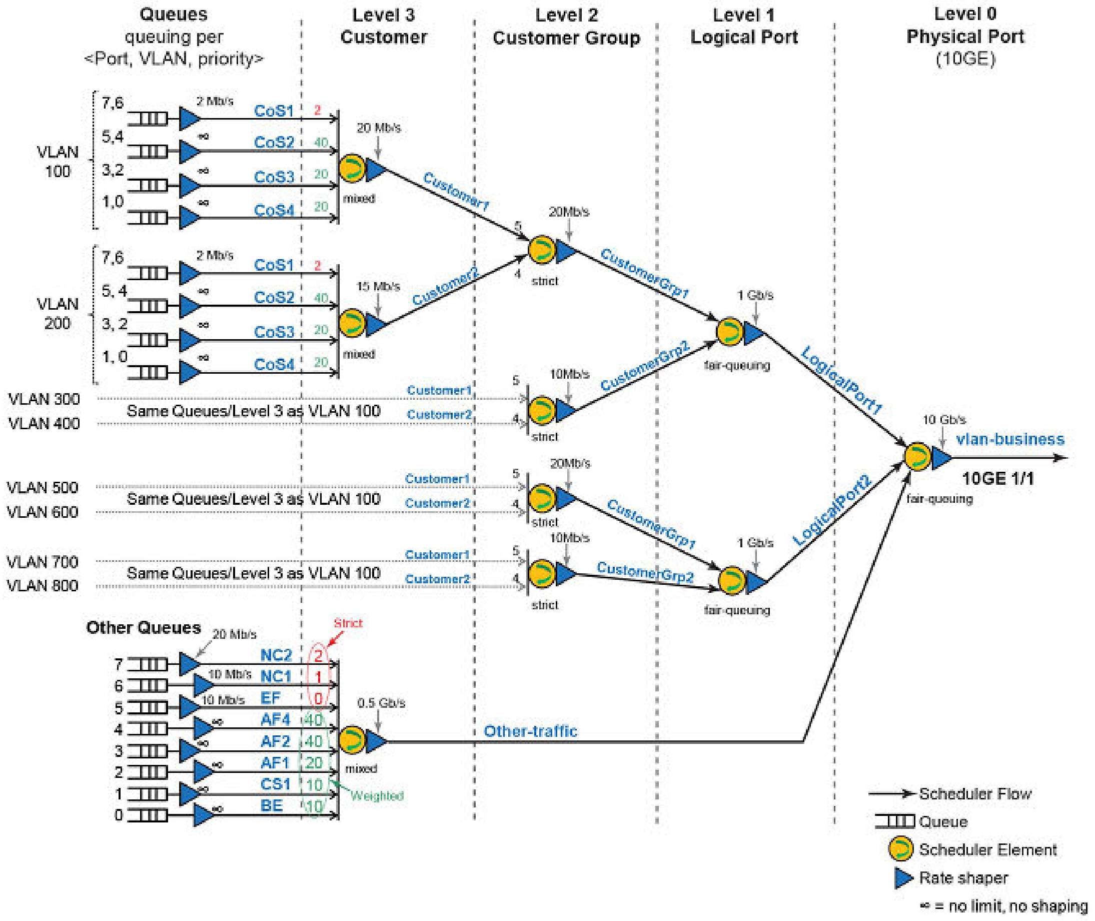
Syntax: clear hqos statistics ethernet *slot/port* **queue** *hqos-scheduler-node-name* **index** *index-number*

Sample configurations

NOTE
All VPLS HQoS traffic in TM will drop after changing the loopback IP address in MPLS configuration.

Local VPLS HQoS example

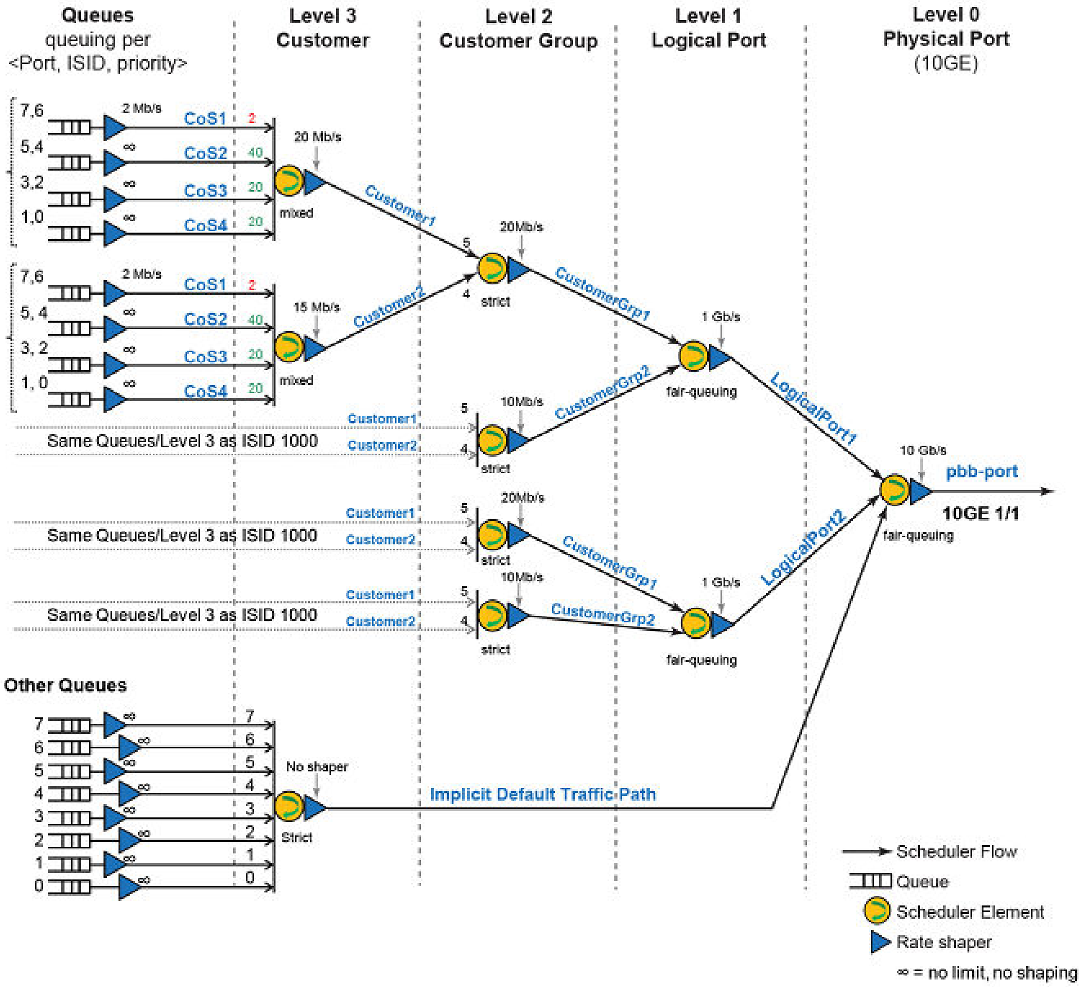
FIGURE 19 VLAN HQoS deployment example



```
hqos scheduler-policy vlan-business level level-0
 shaper-rate 10000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow LogicalPort1 scheduler-input 7 scheduler-policy logical-port-type1
  scheduler-flow LogicalPort2 scheduler-input 6 scheduler-policy logical-port-type1
  scheduler-flow Other-traffic  scheduler-input 5 scheduler-policy other-policy
!
hqos scheduler-policy logical-port-type1 level level-1
 shaper-rate 1000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow CustomerGrp1 scheduler-input 3 scheduler-policy customer-group-
type1
  scheduler-flow CustomerGrp2 scheduler-input 2 scheduler-policy customer-group-type1
!
hqos scheduler-policy customer-group-type1 level level-2
 shaper-rate  20000
 shaper-burst-size 10
 scheduler-type strict
  scheduler-flow Customer1 scheduler-input 3 scheduler-policy customer-type1
  scheduler-flow Customer2 scheduler-input 2 scheduler-policy customer-type1
!
hqos scheduler-policy customer-type1 level  level-3
 shaper-rate 20000
 shaper-burst-size 10
 scheduler-type mixed
  scheduler-flow CoS1 scheduler-input 3 scheduler-policy Q-7-6
  scheduler-flow CoS2 scheduler-input 2 weight 40 scheduler-policy Q-5-4
  scheduler-flow CoS3 scheduler-input 1 weight 20 scheduler-policy Q-3-2
  scheduler-flow CoS4 scheduler-input 0 weight 20 scheduler-policy Q-1-0
!
hqos scheduler-policy other-policy  level level-3
 scheduler-shaper 500000
 scheduler-type-other mixed
  scheduler-flow NC2 scheduler-input 7 scheduler-policy Q-7
  scheduler-flow NC1 scheduler-input 6 scheduler-policy Q-6
  scheduler-flow EF scheduler-input 5 scheduler-policy Q-5
  scheduler-flow AF4 scheduler-input 4 weight 40 scheduler-policy queue-default
  scheduler-flow AF2 scheduler-input 3 weight 40 scheduler-policy queue-default
  scheduler-flow AF1 scheduler-input 2 weight 20 scheduler-policy queue-default
  scheduler-flow CS1 scheduler-input 1 weight 10 scheduler-policy queue-default
  scheduler-flow BE  scheduler-input 0 weight 10 scheduler-policy queue-default
!
hqos queue-policy Q-7-6
    shaper-rate 2000
    shaper-burst-size 10
hqos queue-policy Q-5-4
    shaper-rate 10000000
    shaper-burst-size 10
hqos queue-policy Q-3-2
    shaper-rate 10000000
    shaper-burst-size 10
hqos queue-policy Q-1-0
    shaper-rate 10000000
    shaper-burst-size 10
hqos queue-policy Q-7
   shaper-rate  20000
hqos queue-policy Q-6
    shaper-rate 10000
hqos queue-policy Q-5
    shaper-rate 10000
hqos queue-policy queue-default
    shaper-rate 10000000
    shaper-burst-size 10
!
!
router mpls
 vpls Customer1 1
  vlan 100
   tagged ethe 2/1 ethe 1/1
 vpls Customer2 2
  vlan 200
   tagged ethe 2/1 ethe 1/1
 vpls Customer3 3
  vlan 300
   tagged ethe 2/1 ethe 1/1
 vpls Customer4 4
  vlan 400
   tagged ethe 2/1 ethe 1/1
 vpls Customer5 5
  vlan 500
   tagged ethe 2/1 ethe 1/1
 vpls Customer6 6
  vlan 600
   tagged ethe 2/1 ethe 1/1
 vpls Customer7 7
  vlan 700
   tagged ethe 2/1 ethe 1/1
 vpls Customer8 8
  vlan 800
   tagged ethe 2/1 ethe 1/1
!
!
interface ethernet 1/1
 hqos service-policy output vlan-business
 hqos-map LogicalPort1.CustomerGrp1.Customer1 match vlan 100
 hqos-map LogicalPort1.CustomerGrp1.Customer2 match vlan 200
 hqos-map LogicalPort1.CustomerGrp2.Customer1 match vlan 300
 hqos-map LogicalPort1.CustomerGrp2.Customer2 match vlan 400
 hqos-map LogicalPort2.CustomerGrp1.Customer1 match vlan 500
 hqos-map LogicalPort2.CustomerGrp1.Customer2 match vlan 600
 hqos-map LogicalPort2.CustomerGrp2.Customer1 match vlan 700
 hqos-map LogicalPort2.CustomerGrp2.Customer2 match vlan 800
 hqos-map Other-traffic match other
 hqos-map LogicalPort1.CustomerGrp1.Customer2 shaper-rate 15000
 hqos-map LogicalPort1.CustomerGrp2 shaper-rate 10000
 hqos-map LogicalPort2.CustomerGrp2 shaper-rate 10000
```

PBB HQoS example configuration

FIGURE 20 PBB HQoS deployment example



```
hqos scheduler-policy pbb-port level level-0
 shaper-rate 10000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow LogicalPort1 scheduler-input 7 scheduler-policy logical-port-type1
  scheduler-flow LogicalPort2 scheduler-input 6 scheduler-policy logical-port-type1
!
hqos scheduler-policy logical-port-type1 level level-1
 shaper-rate 1000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow CustomerGrp1 scheduler-input 3 weight 2 scheduler-policy customer-
group-type1
  scheduler-flow CustomerGrp2 scheduler-input 2 weight 1 scheduler-policy customer-
group-type1
!
hqos scheduler-policy customer-group-type1 level level-2
 shaper-rate  20000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow Customer1 scheduler-input 3 scheduler-policy customer-type1
  scheduler-flow Customer2 scheduler-input 2 scheduler-policy customer-type1
!
hqos scheduler-policy customer-type1 level  level-3
 shaper-rate 20000
 shaper-burst-size 10
 scheduler-type strict
  scheduler-flow CoS1 scheduler-input 3 scheduler-policy queue-default
  scheduler-flow CoS2 scheduler-input 2 scheduler-policy queue-default
  scheduler-flow CoS3 scheduler-input 1 scheduler-policy queue-default
  scheduler-flow CoS4 scheduler-input 0 scheduler-policy queue-default
!
hqos queue-policy queue-default
    shaper-rate 10000000
    shaper-burst-size 10
!
!
router mpls
 vpls Customer1 1
  pbb
  vlan 100
   tagged ethe 3/1
  vlan 10 isid 1000
   tagged eth 1/1 ethe 2/8

 vpls Customer2 2
  pbb
  vlan 200
   tagged ethe 3/1
  vlan 10 isid 2000
   tagged eth 1/1 eth 2/8

 vpls Customer3 3
  pbb
  vlan 300
   tagged ethe 3/1
  vlan 10 isid 3000
   tagged eth 1/1 eth 2/8

 vpls Customer4 4
  pbb
  vlan 400
   tagged ethe 3/1
  vlan 10 isid 4000
   tagged eth 1/1 eth 2/8

 vpls Customer5 5
  pbb
  vlan 500
   tagged ethe 3/1
  vlan 10 isid 5000
   tagged eth 1/1 eth 2/8

vpls Customer6 6
  pbb
  vlan 600
   tagged ethe 3/1
  vlan 10 isid 6000
tagged eth 1/1 eth 2/8

 vpls Customer7 7
  pbb
  vlan 700
   tagged ethe 3/1
  vlan 10 isid 7000
   tagged eth 1/1 eth 2/8

 vpls Customer8 8
  pbb
  vlan 800
   tagged ethe 3/1
  vlan 10 isid 8000
   tagged eth 1/1 eth 2/8
!
!interface ethernet 1/1
 hqos service-policy output pbb-port
 hqos-map Logical-Port1.CustomerGrp1.Customer1 match vlan 10 isid 1000
 hqos-map Logical-Port1.CustomerGrp1.Customer2 match vlan 10 isid 2000
 hqos-map Logical-Port1.CustomerGrp2.Customer1 match vlan 10 isid 3000
 hqos-map Logical-Port1.CustomerGrp2.Customer2 match vlan 10 isid 4000
 hqos-map Logical-Port2.CustomerGrp1.Customer1 match vlan 10 isid 5000
 hqos-map Logical-Port2.CustomerGrp1.Customer2 match vlan 10 isid 6000
 hqos-map Logical-Port2.CustomerGrp2.Customer1 match vlan 10 isid 7000
 hqos-map Logical-Port2.CustomerGrp2.Customer2 match vlan 10 isid 8000
```

Scheduler and queue policy configuration templates

Scheduler and queue policy configuration templates are available for creating the HQoS tree.

The configuration does not come into effect till they are bound to a interface supporting HQoS.

Once a policy is bound to an interface, you cannot make any changes to the policy (except shaper rate and burst size). The policy has to be unbound and then make the changes to the policy and rebind it.

The queues can be connected to any of the following levels (3, 2 or 1). It cannot be connected to level 0 directly.

The same policy can be applied to multiple interfaces.

HQoS scheduler policy configuration template

```
  [no] hqos scheduler-policy <scheduler-policy-name> level <level-number> | level-0 |
level-1| level-2 | level-3
    [no][shaper-rate <shaper-rate>]
    [no] [shaper-burst-size <shaper-burst-size>]
    [no]{scheduler-type | scheduler-type-other} <scheduler-type> | strict | weighted
| mixed
    [no]{scheduler-flow <scheduler-flow-name> {scheduler-input <scheduler-input-
value>} | [weight <weight-value>]  scheduler-policy <scheduler-flow-policy-name>}
```

HQoS Queue policy configuration template

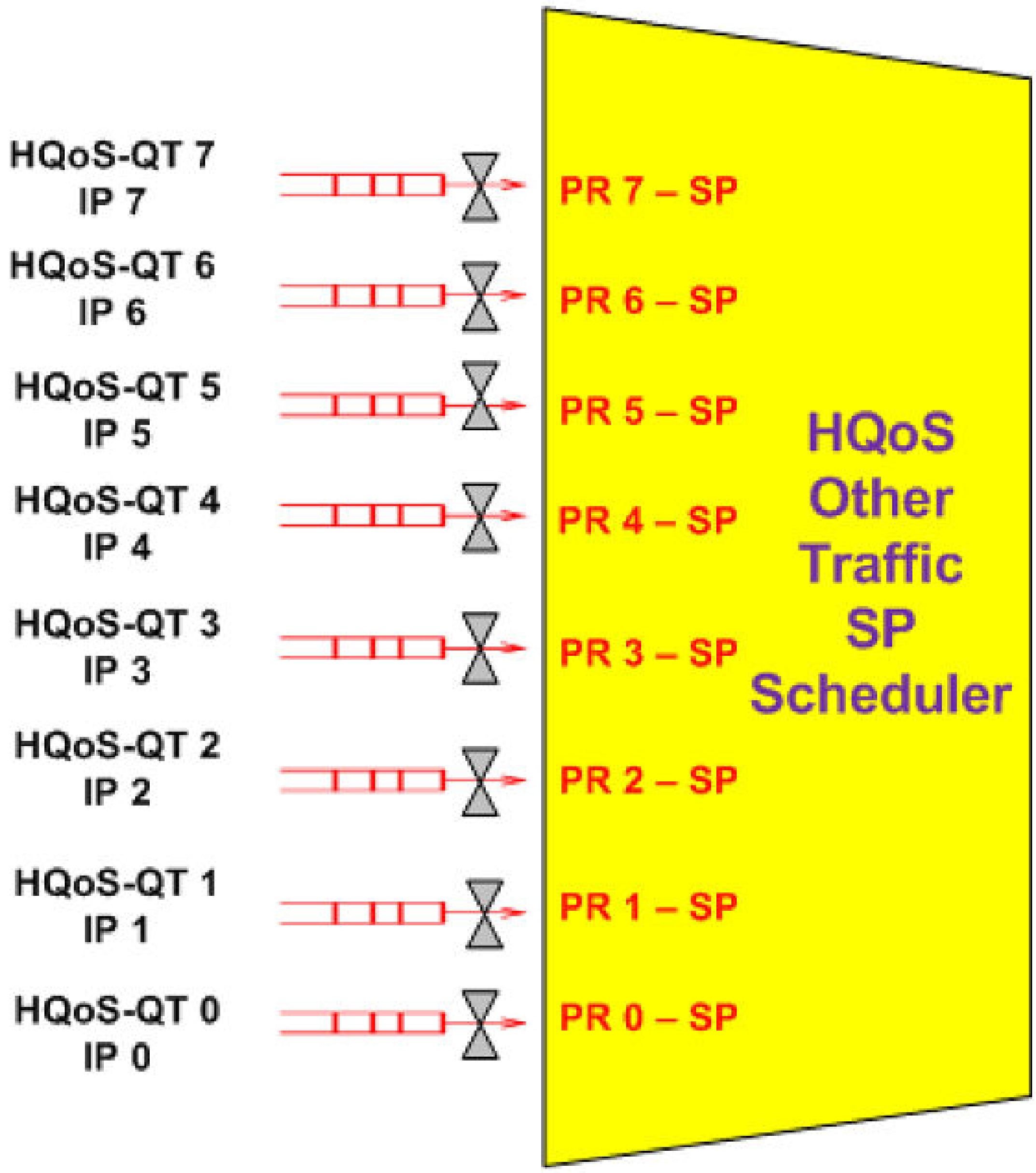
```
[no] hqos queue-policy <queue-name>
    [no] [shaper-rate <shaper-rate>]
    [no] [shaper-burst-size <shaper-burst-size>]
HQOS mapping template.
[no] hqos-map <hqos-scheduler-node-name>
              [no] [shaper-rate <shaper-rate>]
        [no] [shaper-burst-size <shaper-burst-size>]
              [no] [match other] |
              [no] [{match vlan <vlan-num>} | [inner-vlan <inner-vlan-num> | isid
<isid-num> ] ]
HQOS policy binding to an interface
interface ethernet <port >/<slot>
    [no] hqos service-policy output <level-0-scheduler-name>
```

HQoS queue scheduler models

Strict priority (SP)

The figure below is an example of scheduling model for HQoS other traffic. All the 8 scheduler inputs are SP.

Strict priority scheduler (SP)



HQoS-QT stands for hqos-queue-type. The range is <0-7>.

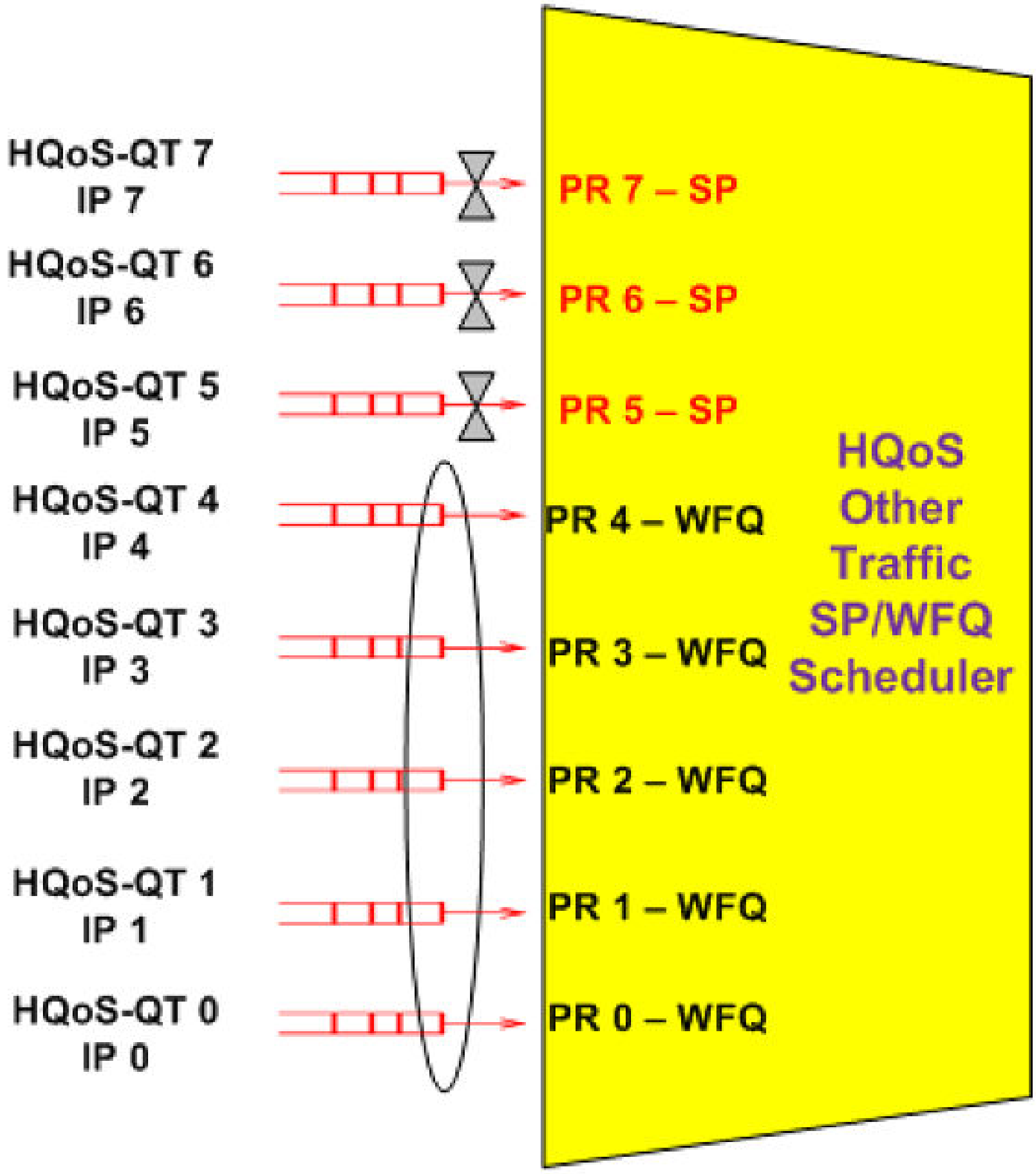
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-7>.

Mixed Strict Priority and Weighted Fair Queue (SP/WFQ)

This figure is an example of mixed SP and WFQ scheduling model for HQoS customer traffic. In this example, the top three scheduler inputs are SP and the bottom five scheduler inputs are WFQ.

FIGURE 21 Mixed Strict Priority and Weighted Fair Queue



The supportable weight range for each input is <1-64>.

HQoS-QT stands for hqos-queue-type. The range is <0-7>.

IP stands for internal priority. The range is <0-7>.

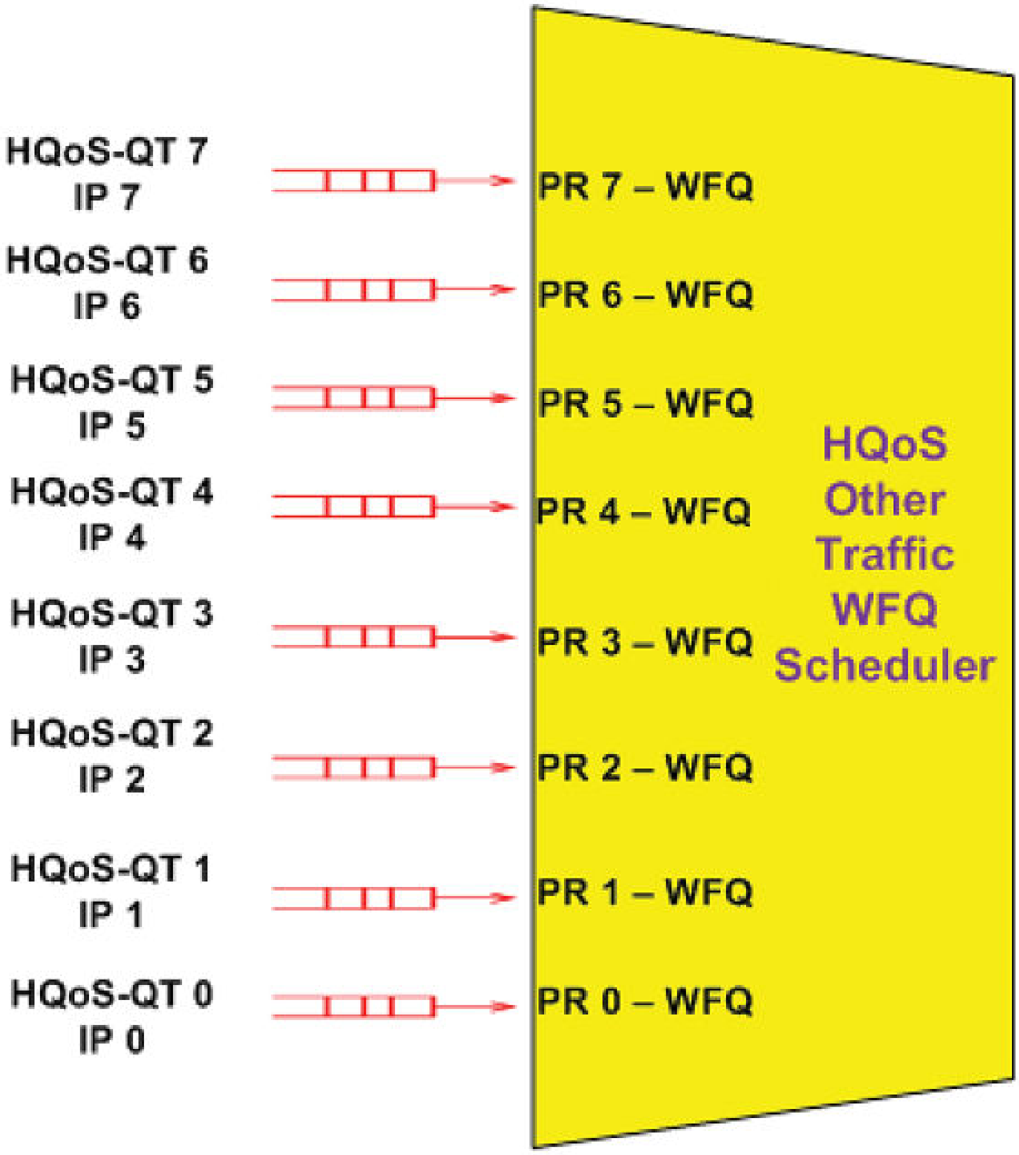
PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-7>.

Weighted Fair Queue and Fair Queue (WFQ/FQ)

This figure is an example the WFQ and FQ scheduling model for HQoS other traffic.

In this example, all 8 scheduler inputs are WFQ. If all 8 scheduler inputs are equal, the scheduling model is Fair Queue (FQ). The supportable weight range for each input is <1-64>.

FIGURE 22 WFQ/FQ scheduling model for HQoS other traffic



HQoS-QT stands for hqos-queue-type. The range is <0-7>.

IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-7>.

QoS Queue Types

The following tables include the system defaults for the HQoS queue-types and internal priority. These 12 queue-types are created in 8x10G modules during the LP initialization.

TABLE 49 HQoS "Other Traffic" queue-type

HQoS "Other Traffic" queue-type	Internal priority	8x10G family buffer-pool	8x10G family default queue size
7	7	Gold	1MB
6	6	Bronze	1MB
5	5	Bronze	1MB
4	4	Bronze	1MB
3	3	Bronze	1MB
2	2	Bronze	1MB
1	1	Bronze	1MB
O	O	Bronze	1MB

TABLE 50 HQoS "Customer Traffic" queue-type

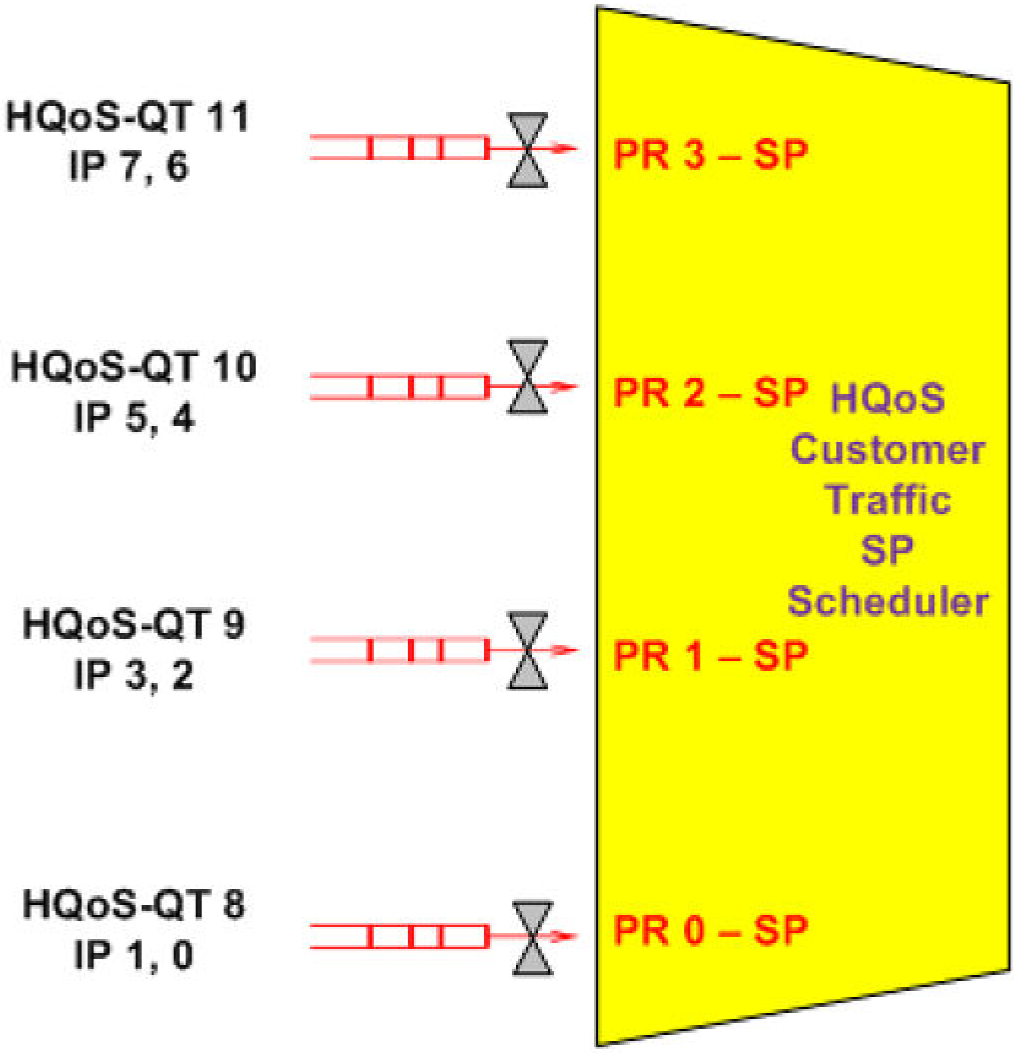
HQoS "Customer Traffic" queue-type	Internal priority	8x10G family buffer-pool	8x10G family default queue size
11	7,6	Gold	1MB
10	5,4	Bronze	1MB
9	3,2	Bronze	1MB
8	0,1	Bronze	1MB

HQoS Queue Schedulers - Customer Traffic

Strict Priority (SP)

This figure depicts the SP scheduling model for HQoS other traffic. All the 4 scheduler inputs are SP.

FIGURE 23 SP scheduling model for HQoS other traffic



HQoS-QT stands for hqos-queue-type. The range is <8-11>.

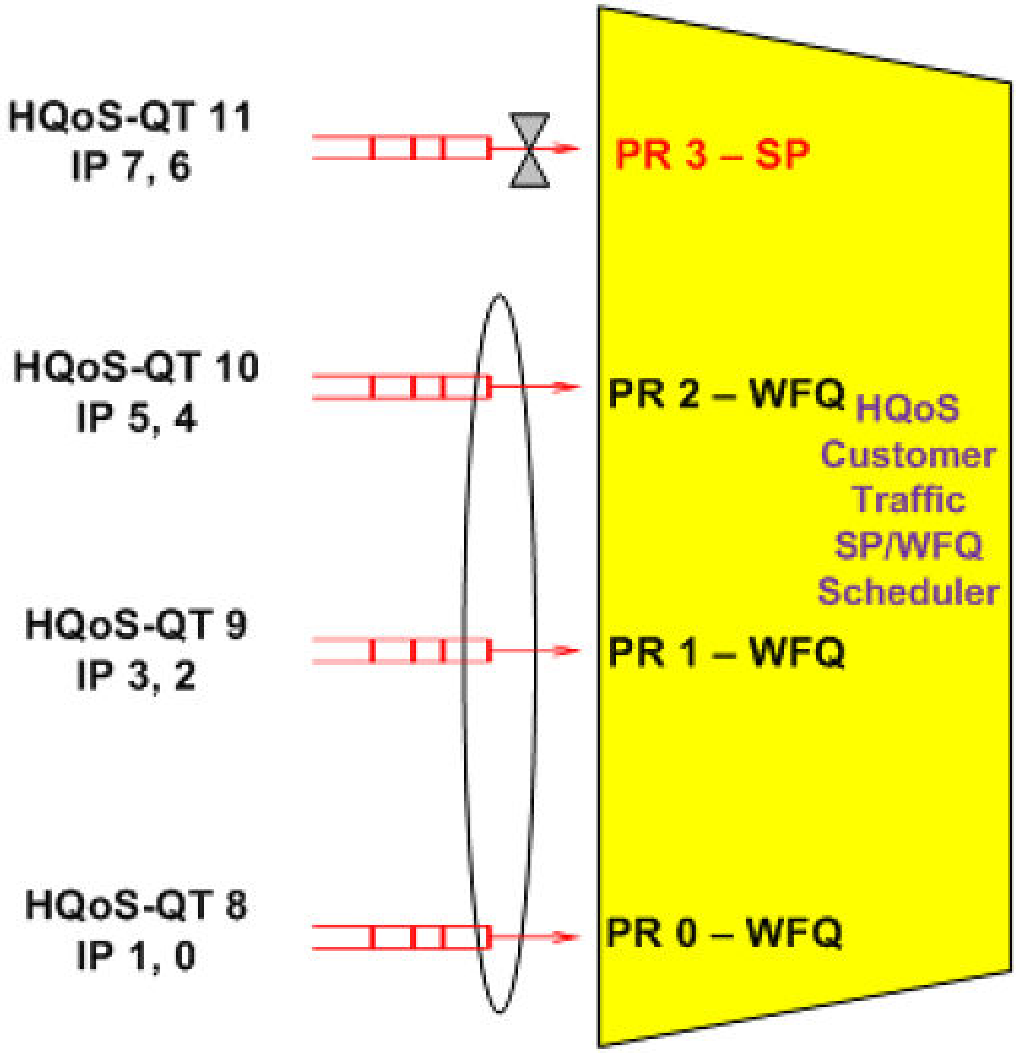
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-3>.

Mixed Strict Priority and Weighted Fair Queue (SP/WFQ)

This figure depicts the mixed SP/WFQ scheduling model for HQoS other traffic. The top scheduler input is SP and the bottom 3 scheduler inputs are WFQ.

FIGURE 24 Mixed Strict Priority and Weighted Fair Queue



The supportable weight range for each input is <1-64>.

HQoS-QT stands for hqos-queue-type. The range is <8-11>.

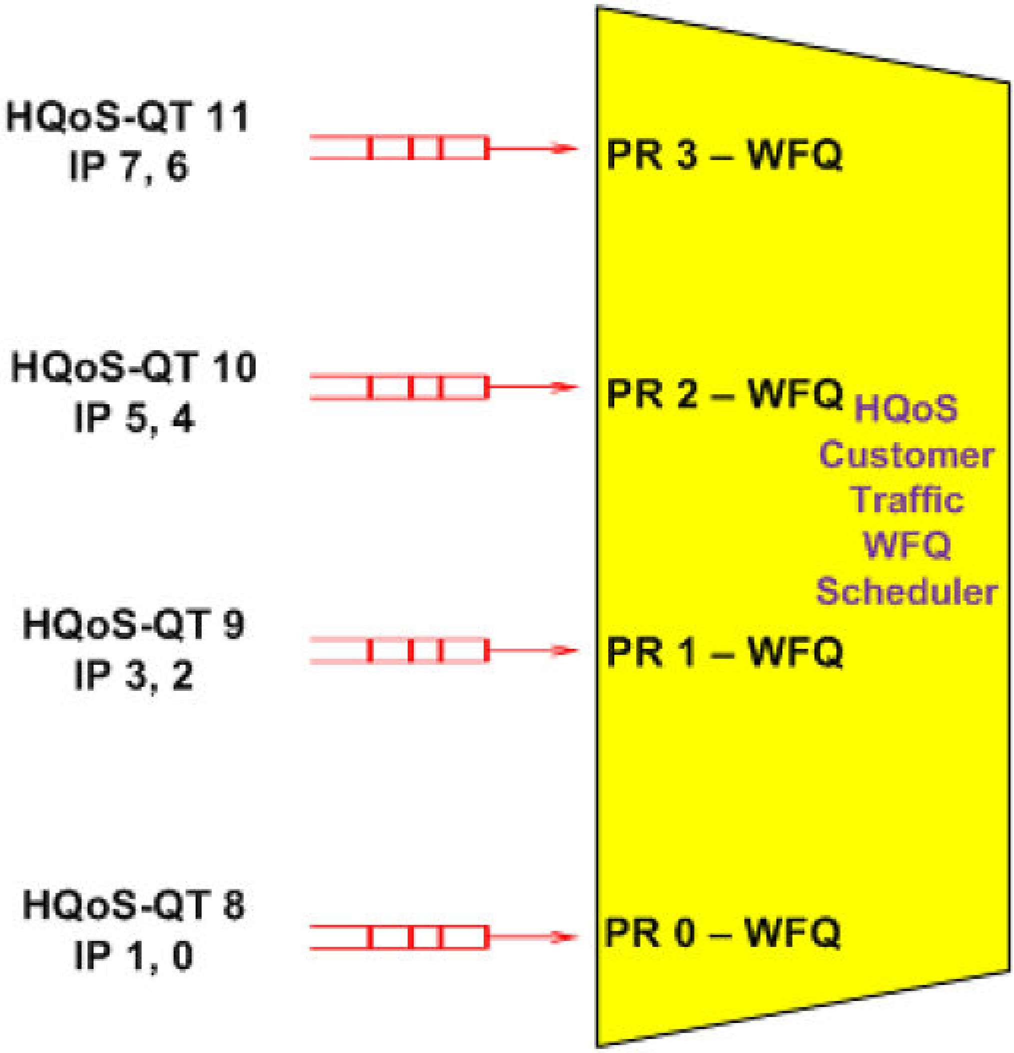
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-3>.

Weighted Fair Queue and Fair Queue (WFQ/FQ)

This figure depicts the mixed Weighted Fair Queue (WFQ) scheduling model for HQoS customer traffic. All the 4 scheduler inputs are WFQ. If all the 4 scheduler inputs are equal, the scheduling model is FQ.

FIGURE 25 Weighted Fair Queue and Fair Queue



The supportable weight range for each input is <1-64>.

HQoS-QT stands for hqos-queue-type. The range is <8-11>.

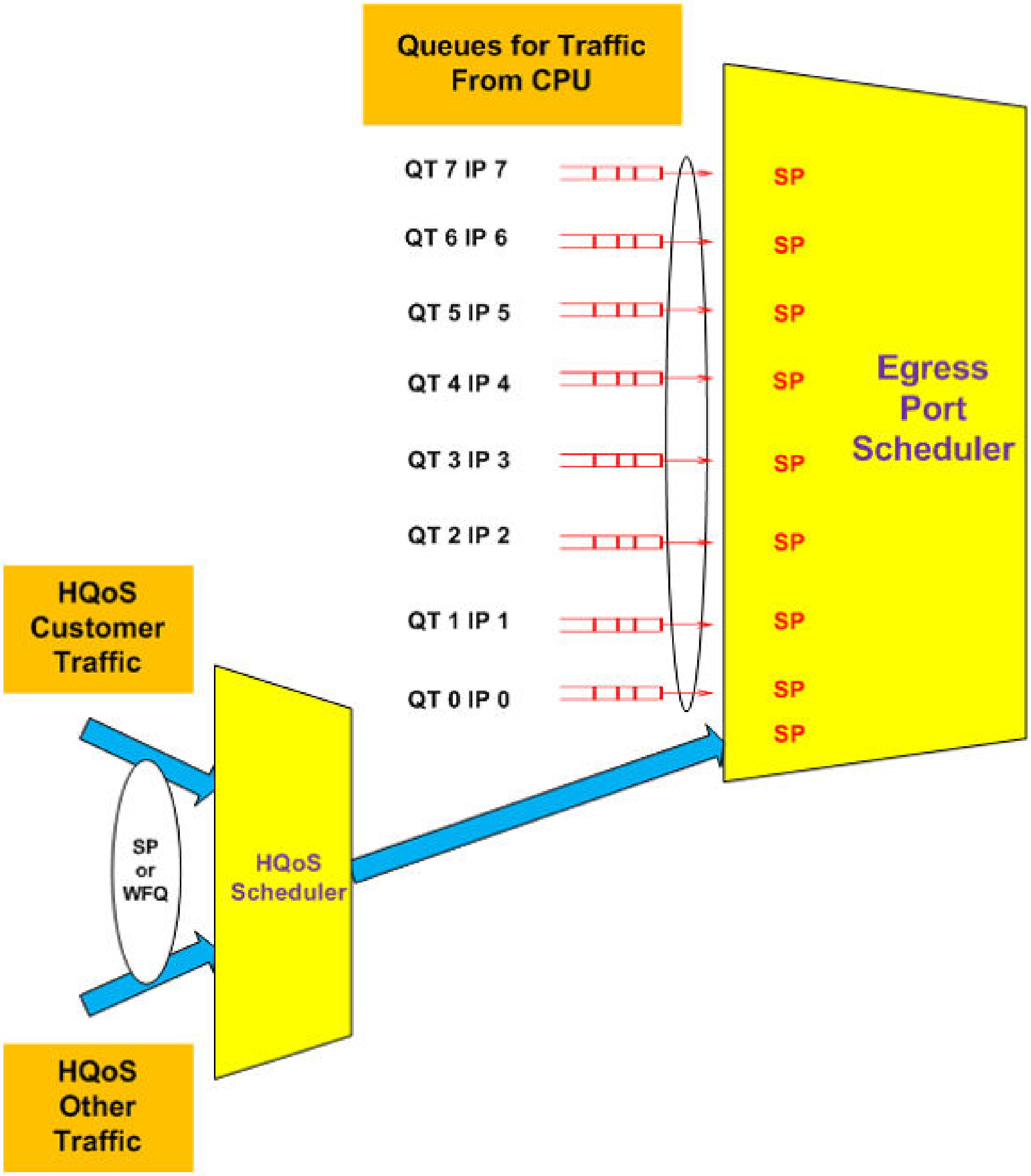
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-3>.

HQoS egress port scheduler

The figure below depicts the Egress Port Scheduler for a port for which HQoS is enabled.

FIGURE 26 HQoS egress port scheduler



The Egress Port Scheduler has 9 SP inputs. The scheduler and queue setup for the first 8 inputs is exactly the same as the current egress port scheduler without HQoS. The queues attached to the first 8 inputs are used for scheduling packets from the CPU, which are sent which includes protocol packets and CPU forwarded packets. The 9th (last) input is used for attaching the HQoS scheduler. The scheduling mechanism between the packets sent from CPU and the level 0 HQoS scheduler is strict priority.

QoS and Traffic Management Commands

arp-guard

Discards all gratuitous ARP and ARP replies for IP addresses not permitted by the specified ARP-guard standard IP access control list (ACL).

Syntax **arp-guard** *arp-guard-access-list-name*

no arp-guard *arp-guard-access-list-name*

Command Default All gratuitous ARP and ARP replies for IP addresses are software forwarded.

Parameters *arp-guard-access-list-name*

ARP packets that do not match the specified ARP guard ACL are dropped by the LP and those which match will be software forwarded.

Modes Interface configuration mode

Usage Guidelines The **no** form of this command removes the ARP-guard filtering of ARP packets.

This command is used in conjunction with the **arp-guard-access-list** command to build a table of allowed IP addresses on the link on which the ARP-guard feature is enabled.

Examples The following example configures the ARP-guard feature to discard all gratuitous ARP and ARP replies for IP addresses that do not match the IP address and MAC address listed in the ACL named arpacl10.

```
device# configure terminal
device(config)# interface ethernet 1/6
device(conf-if-e1000-1/6)# arp-guard-access-list AS201
device(conf-if-e1000-1/6)# permit 10.0.0.2  0001.0002.0003
device(conf-if-e1000-1/6)# arp-guard arpacl10
```

History

Release version	Command history
5.7.00	This command was introduced.

arp-guard-access-list

Creates the ARP guard access list in the global configuration mode.

Syntax

arp-guard-access-list *arp-guard-access-list-name*

no arp-guard-access-list *arp-guard-access-list-name*

Command Default

No ARP guard access list is created.

Parameters

arp-guard-access-list-name

The name of the ARP guard access-list, which contains the list of rules and filters for a specific ARP ACL.

Modes

Global configuration mode.

Usage Guidelines

Examples

The following example creates an ARP guard access list named AS201.

```
device# configure terminal
device(config)# arp-guard-access-list AS201
```

History

Release version	Command history
5.7.00	This command was introduced.

[arp-guard-access-list] permit

Specifies the required set of ACL rules and filters for an associated ARP guard group.

Syntax

permit [vlan-id] *src_ip_address [src_mac_address I any]*

[no] **permit** [vlan-id] *src_ip_address [src_mac_address I any]*

Parameters

vlan-id

The VLAN ID. The VLAN ID range is between 1 and 4090.

src_ip_address

The source IP address.

src_mac_address

The source MAC address.

Modes

ARP-Guard access-list name mode

Usage Guidelines

The **no** form of the command removes the rules and filters for the specific ARP guard group.

Examples

The following command example specifies the required set of ACL rules and filters for the AS201 ARP guard group.

```
Brocadeconfig)#arp-guard-access-list AS201
Brocade(config-arp-guard-access-list-AS201)#permit 100 1.2.3.4 1111.2222.3333
```

History

Release version	Command history
Multi-Service IronWare R05.7.00	This command was introduced.

arp-guard-syslog-timer

Sets the system log timer duration for an ARP guard.

Syntax **arp-guard-syslog-timer** *number_of_seconds*

no arp-guard-syslog-timer *number_of_seconds*

Command Default By default, ARP guard syslog messages for the dropped packets are displayed on the active console for every 60 seconds.

Parameters ***number_of_seconds***

The syslog timer duration that is configurable in seconds. The default value is 60 seconds.

Modes Global configuration mode.

Usage Guidelines The **no** form of the command removes the syslog timer value.

Examples The following command example is used to set the system log timer value at 240 seconds.

```
Brocade(config)# arp-guard-syslog-timer 240
Brocade(config)#sh arp-guard-access-list all
Arp-guard configuration:
!
arp-guard-access-list AS200
!
arp-guard-access-list AS201
permit  any 1.1.1.1 any
permit  any 1.1.1.1 0001.0001.0001
!
arp-guard-syslog-timer 240
!
```

History

Release version	Command history
Multi-Service IronWare R05.7.00	This command was introduced.

show arp-guard statistics

Displays different statistical information of the ARP guard.

Syntax

show arp-guard statistics ethernet { **all** | SLOT/PORT} [**vlan** *vlan-id*]

Parameters

all

All ARP guard port statistics.

SLOT/PORT

Displays statistics specific to a port.

vlan ***vlan-id***

Display statistics specific to a VLAN on a port. The VLAN ID range is between 1 and 4090.

Modes

Privileged configuration mode.

Command Output

The **show arp-guard statistics** command displays the following information:

NOTE
The **show arp-guard statistics** command is used to account the statistics information for primary port of any LAG.

NOTE
For LAG, statistics are supported only for primary port and not on secondary port.

Output field	Description
Port	The port number.
Vlan-id	The VLAN ID.
Total_Arp_pkts_captured	The total number of ARP packets captured.
Total_Arp_pkts_forwarded	The total number of ARP packets forwarded
Total_Arp_pkts_dropped	The total number of ARP packets dropped
LAG : Prim	Displayed only in the show arp-guard statistics ethernet all alone. To denote LAG ID and its Primary port for that LAG associated with all the ARP-Guard enabled ports.

Examples

The following example displays statistics information for all the ports.

```
Brocade(config)# show arp-guard statistics ethernet all
Port         Vlan-id  Total_Arp_pkts_captured  Total_Arp_pkts_forwarded
Total_Arp_pkts_dropped  LAG : Prim
1/1 (Def/Untag)1             0                         0                        0
1/1            3            10000                      9000                     100
1/1            2            10000                      9000                     100
2/1 (Def/Untag)1             0                         0                        0
2/1            2            10000                      9000                     100
2/1            4            10000                      9000                     100
2/1            5            10000                      9000                     100
```

The following example displays statistics information for any individual port.

```
Brocade(config)# show arp-guard statistics ethernet 1/1
Port         Vlan-id  Total_Arp_pkts_captured  Total_Arp_pkts_forwarded
Total_Arp_pkts_dropped  LAG : Prim
1/1 (Def/Untag)1             0                       0                          0
1/1            3            10000                    9000                       100
1/1            2            10000                    9000                       100
```

The following example displays statistics information for particular VLAN of the ARP guard enabled port

```
Brocade# show arp-guard statistics ethernet 1/1 vlan 2
Port           Vlan-id    Total_Arp_pkts_captured  Total_Arp_pkts_forwarded
Total_Arp_pkts_dropped
1/1                     2            10000
9000                                        100
```

History

Release version	Command history
Multi-Service IronWare R05.7.00	This command was introduced.

show arp-guard port-bindings

Displays list of ports associated with the ARP guard access-list.

Syntax **show arp-guard port-bindings** {arp-guard-access-list| **all** }

Parameters **arp-guard-access-list**

Displays port association specific to that particular ARP-Guard access list.

all

Displays port binding association for all the ARP-Guard access-list.

Modes Privileged configuration mode.

Command Output The **show arp-guard port-bindings** command displays the following information:

Output field	Description
Arp-Guard	Displays the name of the ARP-Guard.
Number of Ports	Displays the total number of ports associated with this ARP-Guard.
Port Lists	Displays the list of ports associated with that ARP-Guard.

Examples The following example displays information about the ARP guard port bindings for AS200.

```
Brocade(config-if-e10000-1/8)#show arp-guard port-bindings AS200
Arp-Guard : AS200
Number of Ports : 1
Port Lists : ethe 1/8
```

The following example displays information about the ports associated with ARP guard.

```
Brocade#show arp-guard port-bindings all
Arp-Guard Port Bindings:

Arp-Guard        : ag1
 Number of Ports  : 0

Arp-Guard        : ag2
 Number of Ports  : 2
  Ethe 1/2        Log : Disabled
  Ethe 1/4        Log : Disabled

Arp-Guard        : ag3
 Number of Ports  : 8
  Ethe 1/1        Log : Disabled
  Ethe 2/1        Log : Enabled       Num of violations : Default
  Ethe 2/2        Log : Enabled       Num of violations : 32
  Ethe 2/3        Log : Enabled       Num of violations : 32
  Ethe 2/4        Log : Enabled       Num of violations : 32
  Ethe 2/6        Log : Disabled
  Ethe 3/1        Log : Enabled       Num of violations : Default
  Ethe 4/1        Log : Enabled       Num of violations : Default
```

History

Release version	Command history
Multi-Service IronWare R05.7.00	This command was introduced.

show arp-guard-access-list name

Displays details for a specified ARP guard access list.

Syntax **show arp-guard-access-list name** *arp-guard-access-list* **all**

Parameters ***arp-guard-access-list***

The name of the ARP guard access list.

Modes Privileged configuration mode.

Examples The following example displays information about the ARP guard access list named C5-global-arp.

```
Brocade#sh arp-guard-access-list name C5-global-arp
Arp-Guard : C5-global-arp
Number of rules : 6
Number of Ports : 16
Rules configured
 permit  40 31.0.8.1 0012.f290.7400
 permit  1500 31.0.10.2 0000.0015.0000
 permit  1001 100.0.0.2 0024.38a3.6e00
 permit  20 41.0.100.1 0024.38a3.6e00
 permit  80 51.0.4.2 748e.f874.4900
 permit  any 31.0.11.1 0012.f290.7400
C5(config)#
```

The following example displays information about all the ARP guard access list.

```
Brocade#sh arp-guard-access-list all
Arp-guard configuration:
!
arp-guard-access-list C5-8
!
arp-guard-access-list MCT-A3
 permit  any 31.0.10.2 0000.0300.0000
 permit  any 31.0.10.3 0000.0300.0001
 permit  any 31.0.10.4 0000.0300.0002
 permit  any 31.0.10.5 0000.0300.0003
 permit  any 31.0.11.1 any
 permit  any 31.0.11.2 any
 permit  any 31.0.11.3 any
!
arp-guard-access-list C5-global-arp
 permit  40 31.0.8.1 0012.f290.7400
 permit  1500 31.0.10.2 0000.0015.0000
 permit  1001 100.0.0.2 0024.38a3.6e00
 permit  20 41.0.100.1 0024.38a3.6e00
 permit  80 51.0.4.2 748e.f874.4900
 permit  any 31.0.11.1 0012.f290.7400
!
arp-guard-access-list AS201
 permit  any 1.1.1.1 any
 permit  any 1.1.1.1 0001.0001.0001
!
```

History

Release version	Command history
Multi-Service IronWare R05.7.00	This command was introduced.

clear arp-guard-statistics

Clears the different statistical information of the ARP guard.

Syntax **clear arp-guard statistics ethernet** { **all** | SLOT/PORT} [**vlan** *vlan-id*]

Command Default Clears all statistics related to the ARP guard.

Parameters **all**

All ARP-Guard bound interface's statistics will be cleared.

SLOT/PORT

The specific slot and port value for which you want to clear statistics.

vlan *vlan-id*

VLAN statistics will be cleared specific to that port alone. The VLAN ID range is between 1 and 4090.

Modes Privileged configuration mode.

Usage Guidelines Use the **show arp-guard statistics** command to verify changes after executing the **clear arp-guard statistics** command.

Examples The following example indicates clearing statistics information for all the ports.

```
Brocade# clear arp-guard-statistics all
Brocade# show arp-guard statistics ethernet all
Port         Vlan-id  Total_Arp_pkts_captured  Total_Arp_pkts_forwarded
Total_Arp_pkts_dropped  LAG : Prim
1/1 (Def/Untag)1             0                         0                        0
1/1            3             0                         0                        0
1/1            2             0                         0                        0
2/1 (Def/Untag)1             0                         0                        0
2/1            2             0                         0                        0
2/1            4             0                         0                        0
2/1            5             0                         0                        0
```

The following example indicates clearing statistics information for any individual ports.

```
Brocade#clear arp-guard-statistics ethernet 1/1
Brocade#show arp-guard statistics ethernet 1/1
Port         Vlan-id  Total_Arp_pkts_captured  Total_Arp_pkts_forwarded
Total_Arp_pkts_dropped  LAG : Prim
1/1 (Def/Untag)1             0
0                                              0
1/1            3             0
0                                              0
1/1            2             0
0                                              0
```

The following example indicates clearing statistics information for VLAN ID 2 from port 1/1.

```
Brocade #clear arp-guard-statistics ethernet 1/1 vlan 2
Brocade #show arp-guard statistics ethernet 1/1 vlan 2
Port         Vlan-id  Total_Arp_pkts_captured  Total_Arp_pkts_forwarded
Total_Arp_pkts_dropped  LAG : Prim
1/1            2               0                         0                     0
```

History

Release version	Command history
Multi-Service IronWare R05.7.00	This command was introduced.