



Extreme ONE OS Switching v22.2.1.0 Layer 2 Switching Configuration Guide

Switching, LAG, MLAG, and Bridge Domain Setup

9039425-00 Rev AA
December 2025



Copyright © 2025 Extreme Networks, Inc. All rights reserved.

Legal Notice

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

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

Trademarks

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

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

For additional information on Extreme Networks trademarks, see: <https://www.extremenetworks.com/about-extreme-networks/company/legal/trademarks>

Open Source Declarations

Some software files have been licensed under certain open source or third-party licenses.

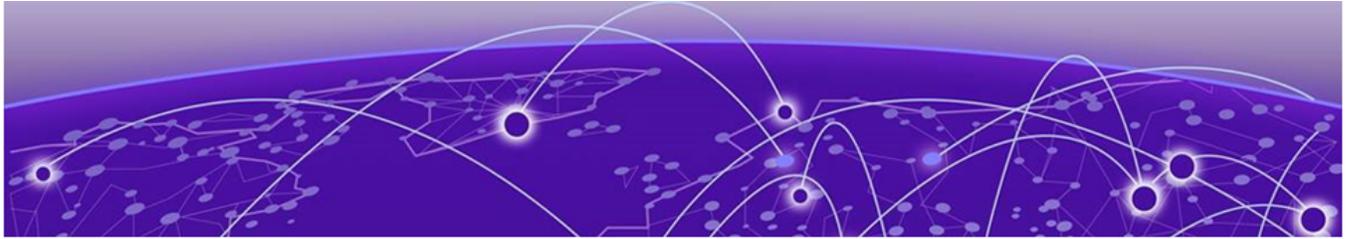
End-user license agreements and open source declarations can be found at: <https://www.extremenetworks.com/support/policies/open-source-declaration/>



Table of Contents

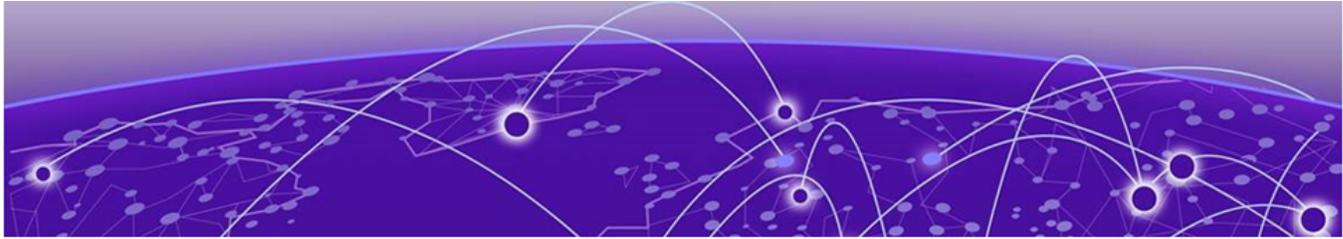
Abstract.....	v
Preface.....	vi
Text Conventions.....	vi
Documentation and Training.....	vii
Open Source Declarations.....	viii
Training.....	viii
Help and Support.....	viii
Subscribe to Product Announcements.....	ix
Send Feedback.....	ix
About This Document.....	10
What's New in This Document	10
Supported Platforms.....	10
Link Aggregation.....	11
Link Aggregation Overview.....	11
LAG Configuration Guidelines.....	12
Basic LAG Configuration.....	12
Configure a Port Channel Interface.....	12
Delete a Port Channel Interface.....	12
Add a Member Port to a Port Channel.....	13
Delete a Member Port from a Port Channel.....	14
Configure the Minimum Number of LAG Member Links.....	14
Dynamic (LACP) Configuration.....	14
Configure LACP.....	15
Configure LACP System Priority	15
Configure the LACP Interval.....	16
Configure the LACP MAC Address.....	17
Configuring System Priority in LACP.....	18
Supported Show and Clear Commands.....	19
Multi-Chassis Link Aggregation (MLAG).....	22
Overview.....	22
MLAG Limitations.....	23
Other Limitations.....	23
Supported Features.....	23
MLAG Terminology.....	24
MLAG Control Plane.....	24
Inter-Switch Link for the Extreme ONE OS SR Devices.....	24
MLAG Node Role Selection in AUTO Role.....	25
MLAG Control Plane Protocol (MLAG CCP).....	25
MLAG Resiliency.....	26
Health Monitoring.....	26

Split-Brain Handling.....	26
Chassis Manager Responsibilities.....	27
FwdHAL Responsibilities.....	27
Out-of-Memory (OOM) Condition.....	27
Uncontrolled Restart of FWD-HAL.....	31
Uncontrolled Restart of Multiple Services.....	32
YANG Modules and CLI Commands.....	32
Using MLAG CLI commands.....	33
Configure MLAG Session on Peer Devices.....	36
MLAG Event Monitoring and Notification.....	43
Supported Notifications.....	43
EXTREME ONE MLAG MIB Definitions.....	44
YANG Module for MLAG Status Retrieval.....	45
Displaying MLAG Addresses in the MAC Address Table for a Bridge Domain.....	46
MLAG Event Log Messages.....	46
RASLogs.....	47
Bridge Domains	49
Bridge Domain Overview.....	49
Bridge Domain Limitations.....	49
Bridge Domain Configuration.....	50
Configure Subinterface.....	50
Configure Bridge Domain in VLAN Mode.....	51
Configure Bridge Domain in Default Mode.....	52
Configure a Static MAC Address.....	53
Configure MAC Learning.....	54
Configure MAC Address Aging.....	56
MAC Movement Detection and Resolution.....	57
MAC Movement Overview.....	57
MAC Movement Detection.....	58
MAC Movement Resolution.....	58
MAC Movement Detection and Resolution Commands.....	60



Abstract

The *Extreme ONE OS Switching Layer 2 Switching Configuration Guide* version 22.2.1.0 details Layer 2 features and procedures, including static and dynamic LAG (IEEE 802.1AX/LACP), MLAG control-plane design, and bridge domain configuration (VLAN and default modes) for tagged, double-tagged, untagged, and untagged-strict sub-interfaces. This guide is intended for intermediate-to-advanced IT professionals.



Preface

Read the following topics to learn about:

- The meanings of text formats used in this document.
- Where you can find additional information and help.
- How to reach us with questions and comments.

Text Conventions

Unless otherwise noted, information in this document applies to all supported environments for the products in question. Exceptions, like command keywords associated with a specific software version, are identified in the text.

When a feature, function, or operation pertains to a specific hardware product, the product name is used. When features, functions, and operations are the same across an entire product family, such as Extreme Networks switches, the product is referred to as *the switch*.

Table 1: Notes and warnings

Icon	Notice type	Alerts you to..
	Tip	Helpful tips and notices for using the product
	Note	Useful information or instructions
	Important	Important features or instructions
	Caution	Risk of personal injury, system damage, or loss of data
	Warning	Risk of severe personal injury

Table 2: Text

Convention	Description
screen displays	This typeface indicates command syntax, or represents information as it is displayed on the screen.
The words <i>enter</i> and <i>type</i>	When you see the word <i>enter</i> in this guide, you must type something, and then press the Return or Enter key. Do not press the Return or Enter key when an instruction simply says <i>type</i> .
Key names	Key names are written in boldface, for example Ctrl or Esc . If you must press two or more keys simultaneously, the key names are linked with a plus sign (+). Example: Press Ctrl+Alt+Del
<i>Words in italicized type</i>	Italics emphasize a point or denote new terms at the place where they are defined in the text. Italics are also used when referring to publication titles.
NEW!	New information. In a PDF, this is searchable text.

Table 3: Command syntax

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

Documentation and Training

Find Extreme Networks product information at the following locations:

[Current Product Documentation](#)

[Release Notes](#)

[Hardware and Software Compatibility](#) for Extreme Networks products

[Extreme Optics Compatibility](#)

[Other Resources](#) such as articles, white papers, and case studies

Open Source Declarations

Some software files have been licensed under certain open source licenses. Information is available on the [Open Source Declaration](#) page.

Training

Extreme Networks offers product training courses, both online and in person, as well as specialized certifications. For details, visit the [Extreme Networks Training](#) page.

Help and Support

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

Extreme Portal

Search the GTAC (Global Technical Assistance Center) knowledge base; manage support cases and service contracts; download software; and obtain product licensing, training, and certifications.

The Hub

A forum for Extreme Networks customers to connect with one another, answer questions, and share ideas and feedback. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.

Call GTAC

For immediate support: (800) 998 2408 (toll-free in U.S. and Canada) or 1 (408) 579 2800. For the support phone number in your country, visit www.extremenetworks.com/support/contact.

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

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

Subscribe to Product Announcements

You can subscribe to email notifications for product and software release announcements, Field Notices, and Vulnerability Notices.

1. Go to [The Hub](#).
2. In the list of categories, expand the **Product Announcements** list.
3. Select a product for which you would like to receive notifications.
4. Select **Subscribe**.
5. To select additional products, return to the **Product Announcements** list and repeat steps 3 and 4.

You can modify your product selections or unsubscribe at any time.

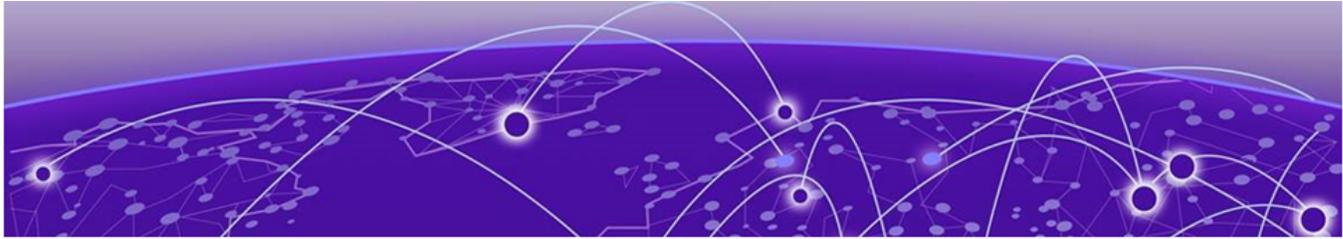
Send Feedback

The User Enablement team at Extreme Networks has made every effort to ensure that this document is accurate, complete, and easy to use. We strive to improve our documentation to help you in your work, so we want to hear from you. We welcome all feedback, but we especially want to know about:

- Content errors, or confusing or conflicting information.
- Improvements that would help you find relevant information.
- Broken links or usability issues.

To send feedback, email us at Product-Documentation@extremenetworks.com.

Provide as much detail as possible including the publication title, topic heading, and page number (if applicable), along with your comments and suggestions for improvement.



About This Document

[What's New in This Document](#) on page 10

[Supported Platforms](#) on page 10

What's New in This Document

The following table describes the information added to this guide for Extreme ONE OS Switching, release 22.2.1.0:

Feature	Description	Link
Bridge domains	Updated the topics "Configure Subinterface," "Configure Bridge Domain in VLAN Mode," and "Configure Bridge Domain in Default Mode."	<ul style="list-style-type: none">• Bridge Domain Configuration on page 50

For additional information, refer to the *Extreme ONE OS Switching Release Notes*.

Supported Platforms

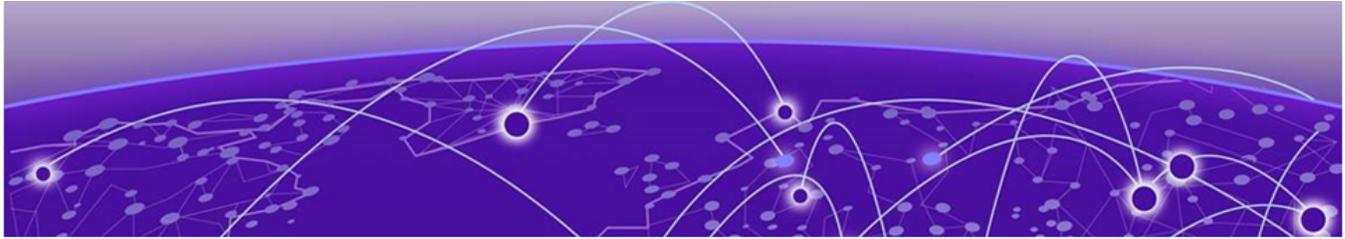
Extreme ONE OS Switching 22.2.1.0 supports Extreme 8520, Extreme 8720, Extreme 8730, and Extreme 8820 hardware platforms.



Note

Although many software and hardware configurations are tested and supported for this release, documenting all possible configurations and scenarios is beyond this document's scope.

For information about other releases, see the documentation for those releases.



Link Aggregation

[Link Aggregation Overview](#) on page 11

[Basic LAG Configuration](#) on page 12

[Dynamic \(LACP\) Configuration](#) on page 14

[Supported Show and Clear Commands](#) on page 19

Link Aggregation Overview

Link aggregation enables you to bundle multiple physical Ethernet links into a single port channel, providing enhanced performance, redundancy, and availability.

We also refer to port channels as link aggregation groups (LAGs). A LAG is considered a single link by connected devices, the Spanning Tree Protocol, IEEE 802.1Q VLANs, and so on. When one physical link in the LAG fails, the other links stay up. A small drop in traffic is experienced when one link fails.

When queuing traffic from multiple input sources to the same output port, all input sources are given the same weight, regardless of whether the input source is a single physical link or a port channel.

The benefits of link aggregation are as follows:

- Increased bandwidth (The logical bandwidth can be dynamically changed as the demand changes.)
- Increased availability
- Load sharing
- Rapid configuration and reconfiguration

Each LAG consists of the following components:

- Links of the same speed.
- A MAC address that is different from the MAC addresses of the LAG's individual member links.
- An interface index for each link to identify the link to the neighboring devices.
- An administrative key for each link. Only the links with the same administrative key value can be aggregated into a LAG. On each link configured to use Link Aggregation Control Protocol (LACP), LACP automatically configures an administrative key value equal to the port channel identification number.

Two LAG types are supported:

- Static LAGs—In static link aggregation, links are added into a LAG without exchanging control packets between the partner systems. The distribution and collection of frames on static links is determined by the operational status and administrative state of the link.
- Dynamic LAGs—Dynamic link aggregation uses Link Aggregation Control Protocol (LACP) to negotiate the links included in a LAG. Typically, two partner systems sharing multiple physical Ethernet links can aggregate a number of those physical links using LACP. LACP creates a LAG on both partner systems and identifies the LAG by the LAG ID. All links with the same administrative key, and all links that are connected to the same partner switch become members of the LAG. LACP continuously exchanges LACP protocol data units (PDUs) to monitor the health of each member link.

LAG Configuration Guidelines

When implementing port channels, consider the following:

- You can associate a link with only one port channel.

Basic LAG Configuration

The topics in this section configure both static and dynamic (LACP) LAG implementations.

Configure a Port Channel Interface

Follow this procedure to create a port channel interface at the global configuration mode.

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
```

2. Enter the **interface port-channel** command to create a new port channel interface.

```
device(config)# interface port-channel 1
```

Delete a Port Channel Interface

Follow this procedure to delete a port-channel interface and all member interfaces.

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
```

2. To delete a port-channel interface, enter the **no interface port-channel** command.

```
device(config)# no interface port-channel 1
```

Add a Member Port to a Port Channel

Follow this procedure to add a member port to a specific port channel interface. If the port channel does not already exist, this task creates the port channel and also adds a physical interface to it.

1. Access global configuration mode.

```
device# configure terminal
```

2. Add a port channel.

```
device(config)# interface port-channel 1
```

The range is 1 to 255.

3. Access global configuration mode again.

```
device(config-if-po-1)# exit
```

4. Access interface configuration mode for the physical interface that you want to add to the port channel.

```
device(config)# interface ethernet 0/1
```

5. Enable link aggregation on the physical interface and add it to the port channel.

- (For *active* mode) Enable link aggregation on the physical interface and add it to the port channel to enable LACP unconditionally.

```
device(config-if-eth-0/1)# channel-group 1 mode active
```

- (For *passive* mode) Enable link aggregation on the physical interface and add it to the port channel to enable LACP when another LACP device is detected.

```
device(config-if-eth-0/1)# channel-group 1 mode passive
```

- (For *static* mode) Enable link aggregation on the physical interface and add it to the port channel to enable static link aggregation without LACP, which prevents channel formation with other ports that are in active or passive mode.

```
device(config-if-eth-0/1)# channel-group 1 mode on
```

The following example displays the LAG configuration that is running currently on the device. In this example, port channel number 1 is configured, and Ethernet interface 0/1 is added to the port channel in active mode:

```
device# show running-config lACP

interface port-channel 1
  no shutdown
  subinterface vlan 1
    ipv4 address 10.x.x.x/24
    ipv6 address 1001:x:x:x::1/64
  !
!
interface ethernet 0/1
  no shutdown
  channel-group 1 mode active
!
device#
```

Delete a Member Port from a Port Channel

Follow this procedure to delete a member port from a port channel interface at the interface configuration level.

1. Access interface configuration mode for the physical interface that you want to delete from the port channel.

```
device(config)# interface ethernet 0/1
```

2. Delete a port from the port channel interface.

```
device(config-if-eth-0/1)# no channel-group
```

Configure the Minimum Number of LAG Member Links

Follow this procedure to configure the minimum number of LAG member links allowed in the LACP bundle.

This configuration allows a port-channel to operate at a certain minimum bandwidth at all times. If the bandwidth of the port-channel drops below the minimum number, then the port-channel is declared operationally DOWN even though it has operationally UP members.

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
device(config)#
```

2. Enter the **interface port-channel** command at the global configuration level.

```
device(config)# interface port-channel 30
device(config-if-po-30)#
```

3. Configure the minimum number of LAG member links at the port-channel interface configuration mode.

```
device(config-if-po-30)# lacp min-links 5
```

The number of links ranges from 1 to 64. The default minimum number of links is 1.



Note

Always configure min-links identically on both ends of the LAG. This prevents asymmetric forwarding and ensures clean failover behavior.

Dynamic (LACP) Configuration

Link Aggregation Control Protocol (LACP) is an IEEE 802.1AX standards-based protocol that allows two partner systems to dynamically negotiate attributes of physical links between them to form port-channels.

If LACP determines that a link can be aggregated into a LAG, LACP puts the link into the LAG. All links in a LAG inherit the same administrative characteristics.

LACP operates in two modes:

- *Active mode*—LACP initiates protocol data unit (PDU) exchanges, regardless of whether the partner system sends LACP PDUs.

- *Passive mode*—LACP responds to PDUs initiated by its partner system, but does not initiate the LACP PDU exchange.

The LACP process collects and distributes Ethernet frames. The collection and distribution process implements:

- Inserting and capturing control LACP protocol data units (PDUs).
- Restricting the traffic of a given conversation to a specific link.
- Load-balancing links.
- Handling dynamic changes in LAG membership.

On each port, link aggregation control:

- Maintains configuration information to control port aggregation.
- Exchanges configuration information with other devices to form LAGs.
- Attaches ports to and detaches ports from the aggregator when they join or leave a LAG.
- Enables or disables an aggregator's frame collection and distribution functions.

Configure LACP

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
```

2. Enable LACP globally.

```
device(config)# protocol lacp
```

3. Disable LACP globally.

```
device(config)# no protocol lacp
```

Configure LACP System Priority

You configure LACP system priority on each device running LACP. LACP uses the system priority with the device MAC address to form the system ID and also during negotiation with other devices. The system priority value must be a number in the range of 1 through 65535. The higher the number, the lower the priority. The default priority is 32768.

1. Enter the **configure terminal** command to access global configuration mode.

```
device# configure terminal
```

2. Specify the LACP system priority.

```
device(config)# lacp system-priority 25000
```

3. To reset the system priority to the default value.

```
device(config)# no lacp system-priority
```

Configure the LACP Interval

You can configure the protocol data unit (PDU) interval for a port that is a member of a link aggregation group (LAG). This is the interval at which LACP control packets are sent to an LACP supported interface. The default interval is 30 seconds.

The LACP timeout is calculated on the LACP interval (either fast or slow). For the fast setting, the timeout is $3 \times 1 = 3$ seconds. For the slow setting, the timeout is $30 \times 3 = 90$ seconds.

1. Access global configuration mode.

```
device# configure terminal
```

2. Specify the Ethernet interface for which you want to configure the interval.

```
device(config)# interface ethernet 0/1
```

3. Configure the interval on the interface.

- To set a fast interval (one PDU per second):

```
device(config-if-eth-0/1)# lacp interval fast
```

- To set a slow interval (one PDU per 30 seconds):

```
device(config-if-eth-0/1)# lacp interval slow
```

- To restore the default interval (also one PDU per 30 seconds):

```
device(config-if-eth-0/1)# no lacp interval
```

The following example adds Ethernet interface 0/1 to a LAG that has port channel number 10, specifies active LACP mode on the interface, and sets the interval of the interface to one PDU per second:

```
device# configure terminal
device(config)# interface ethernet 0/1
(config-if-eth-0/1)# channel-group 10 mode active
(config-if-eth-0/1)# lacp interval fast
(config-if-eth-0/1)#
```

The following example displays the LAG configuration that is running currently on the device. In this example, Ethernet interface 0/1 is added to a LAG in active mode and with a fast interval (one PDU per second):

```
device# show running-config lacp

protocol lacp
lacp system-priority 32768
!
interface port-channel 10
lacp min-links 1
no shutdown
!
interface ethernet 0/1
channel-group 10 mode active
lacp interval fast
no shutdown
!
device#
```

The following example adds Ethernet interface 0/2 to a LAG that has port channel number 20, specifies active LACP mode on the interface, and sets the interval of the interface to one PDU per 30 seconds:

```
device# configure terminal
device(config)# interface ethernet 0/2
(config-if-eth-0/2)# channel-group 20 mode active
(config-if-eth-0/2)# lacp interval slow
(config-if-eth-0/2)#
```

The following example displays the LAG configuration that is running currently on the device. In this example, Ethernet interface 0/2 is added to a LAG in active mode and with a show interval (one PDU per 30 second):

```
device# show running-config lacp

protocol lacp
lacp system-priority 32768
!
interface port-channel 20
lacp min-links 1
no shutdown
!
interface ethernet 0/2
channel-group 20 mode active
lacp interval fast
no shutdown
!
device#
```

Configure the LACP MAC Address

You can configure the MAC address of a link aggregation group (LAG).

1. Access global configuration mode.

```
device# configure terminal
```

2. Specify the port channel interface for which you want to configure the MAC address.

```
device(config)# interface port-channel 1
```

3. After configuring a port channel, validate the port channel health and statistics information.

```
device# show counters interface port-channel 101
Interface Statistics: port-channel 101
Carrier Transitions: 1
LastClear: 0s
Input:
    Total pkts: 1477460957
Broadcast pkts: 87289588
Discard pkts: 0
Errors pkts: 0
FCS Errors: 0
MCast pkts: 87378080
    Octets: 1126773024923
UCast pkts: 1302793289
    Runt pkts: 0
    CRC Errors: 0
Input Distribution:
    64 byte pkts: 5868861
    65-127 byte pkts: 50932300
```

```

    128-255 byte pkts: 573604220
    256-511 byte pkts: 66253377
    512-1023 byte pkts: 347495781
    1024-1518 byte pkts: 211842668
        Jumbo pkts: 221463750
Out:
    Total pkts: 2416872288
    Broadcast pkts: 522284647
    Discard pkts: 0
    Errors pkts: 0
    MCast pkts: 272962830
        Octets: 2360836407771
    UCast pkts: 1621624811
Rate Info:
    Input: 116.925313 Mbits/sec, 20726 pkts/sec 0.29% of line-rate
    Output: 288.130020 Mbits/sec, 35007 pkts/sec 0.72% of line-rate

```

4. Configure the LACP system MAC address.

```
device(config-if-po-1)# lacp system-mac 66:fc:1d:1f:5b:85
```

5. Reset the LACP system MAC address to the default value. This is a MAC address whose last octet is one more than that of the MAC address of the immediately previous interface.

```
device(config-if-po-1)# no lacp system-mac
```

The following example enables LACP globally on the device and configures a LAG with port channel number 10 and a system MAC address of 01:23:45:67:89:ab:

```

device# configure terminal
device(config)# protocol lacp
device(config)# interface port-channel 10
device(config-if-po-10)# lacp system-mac 01:23:45:67:89:ab
device(config-if-po-10)#

```

The following example displays the LAG configuration that is running currently on the device. In this example, LACP is enabled globally, a LAG with port channel number 10 and a system MAC address of 01:23:45:67:89:ab is configured, and Ethernet interface 0/29 is added to the LAG in active mode:

```

device# show running-config lacp

protocol lacp
interface port-channel 10
lacp system-mac 01:23:45:67:89:ab
no shutdown
!
interface ethernet 0/29
channel-group 10 mode active
no shutdown
!
device#

```

Configuring System Priority in LACP

In LACP, the system priority determines which switch or device takes the lead during link aggregation negotiations. You can set a value that influences which switch takes the lead in a multi-device setup.

The default is 32768, and it can range from 1 to 65535. Ensure that the value is the same across the switch.

To set or unset LACP system-priority, use the following command:

```
device(config-if-po-101)# lacp
  min-links          Set minimum number of links
  system-mac        Set LACP system-mac
  system-priority    Set LACP system priority

device(config-if-po-101)# lacp system-priority
(1-65535) Value from 1-65535
DUT5(config-if-po-101)# lacp system-priority
```

Supported Show and Clear Commands

- **show interface port-channel:** Displays details of port channels.

```
device# show interface port-channel
  IFNAME Po Value from 1-255
  brief  brief
device# show interface port-channel 101

device# show interface port-channel 101
port-channel 101 is up
  MTU 9216 Bytes
  IfIndex 0x4000065
  Mac address is 88:7e:25:d3:da:14
  Port mode is Full Duplex, 40G
  MinLinks is 1
  LagType is LACP
  Active Members in this channel: Eth 0/3
  Members in this channel: Eth 0/3
Statistics
  Carrier Transitions: 1
  LastClear: 0s
Input:
  Broadcast pkts: 87225518
  Discard pkts: 0
  Errors pkts: 0
  FCS Errors: 0
  MCast pkts: 87313943
  Octets: 1126070215752
  UCast pkts: 1301896993
  Unknown Protocols: 0
Out:
  Broadcast pkts: 521900301
  Discard pkts: 0
  Errors pkts: 0
  MCast pkts: 272758354
  Octets: 2359154212029
  UCast pkts: 1620447729

device# show interface port-channel brief
Flags: M - Redundant Management P - Performance-Path
Number of interfaces 21
Port      Mtu      Admin-State Oper-State  Speed    Ifindex    Description
-----
Po 10     9216     UP          UP          75G     0x400000a  ISL_Underlay_PO
Po 53     9216     UP          UP          75G     0x4000035  TO-DUT3
Po 54     9216     UP          UP          75G     0x4000036  TO-DUT4
```

Po 101	9216	UP	UP	40G	0x4000065	Port-Channel 101
Po 102	9216	UP	UP	20G	0x4000066	Port-Channel 102
Po 127	9216	UP	DOWN	0G	0x400007f	Port-Channel 127
Po 151	9216	UP	UP	100G	0x4000097	Port-Channel 151
Po 152	9216	UP	UP	30G	0x4000098	Port-Channel 152
Po 153	9216	UP	UP	20G	0x4000099	Port-Channel 153
Po 154	9216	UP	UP	10G	0x400009a	Port-Channel 154
Po 155	9216	UP	UP	10G	0x400009b	Port-Channel 155
Po 156	9216	UP	UP	40G	0x400009c	Port-Channel 156
Po 157	9216	UP	UP	100G	0x400009d	Port-Channel 157
Po 158	9216	UP	UP	10G	0x400009e	Port-Channel 158
Po 159	9216	UP	UP	10G	0x400009f	Port-Channel 159

- **show counters interface port-channel:** Displays statistics for the port channel.

```

device# show counters interface port-channel 101
Interface Statistics: port-channel 101
  Carrier Transitions: 1
    LastClear: 0s
Input:
  Total pkts: 1477460957
  Broadcast pkts: 87289588
  Discard pkts: 0
  Errors pkts: 0
    FCS Errors: 0
  MCast pkts: 87378080
    Octets: 1126773024923
  UCast pkts: 1302793289
  Runt pkts: 0
  CRC Errors: 0
Input Distribution:
  64 byte pkts: 5868861
  65-127 byte pkts: 50932300
  128-255 byte pkts: 573604220
  256-511 byte pkts: 66253377
  512-1023 byte pkts: 347495781
  1024-1518 byte pkts: 211842668
  Jumbo pkts: 221463750
Out:
  Total pkts: 2416872288
  Broadcast pkts: 522284647
  Discard pkts: 0
  Errors pkts: 0
  MCast pkts: 272962830
    Octets: 2360836407771
  UCast pkts: 1621624811
Rate Info:
  Input: 116.925313 Mbits/sec, 20726 pkts/sec 0.29% of line-rate
  Output: 288.130020 Mbits/sec, 35007 pkts/sec 0.72% of line-rate

```

- **show counters lacp:** Displays statistics for member ports of dynamic link aggregation groups (LAGs).

```

device# show counters lacp
Port          in-pkts  out-pkts  TxErr  RxErr  unknownErr  LACPErr
-----
Channel group: 10
ethernet 0/7:1    393      2724      0       0       0           0
ethernet 0/7:2    401      2723      0       0       0           0

```

```

ethernet 0/7:3      403      2723      0      0      0      0
ethernet 0/7:4      396      2723      0      0      0      0

```



Note

The LACP microservice fetches counter statistics for out-pkts from the kernel every 10 sec. This means that counters are refreshed every 10 seconds. In-Pkts are updated every second.

- **show lacp system-identifier:** Displays the unique identifier assigned to a LAG.

```

device# show lacp system-identifier
System ID: 0x8000, 00:04:96:d6:83:e0

```

- **show lacp interface ethernet x/x:** Display statistics for specific member ports of dynamic LAGs.

```

device# show lacp interface ethernet 0/7:2
interface Eth 0/7:2 is up
  Channel group is 10 port channel is Po10
  PDUs sent: 2943
  PDUs rcvd: 614
  LACP Rx errors: 0
  LACP Tx errors: 0
  LACP unknown errors: 0
  LACP errors: 0
Local Port: Eth 0/7:2  MAC Address = 00:04:96:d6:83:e0
System Identifier = 80:00:00:04:96:d6:83:e0
Port Identifier = 0x8000, 0x207
Operational key = 10
LACP_Activity = active
LACP_Timeout = short Timeout (1s)
Synchronization = IN_SYNC
Collecting = true
Distributing = true

Partner information
  Partner-id = 80:00:00:04:96:d6:55:1c
  Partner-key = {10, 519}

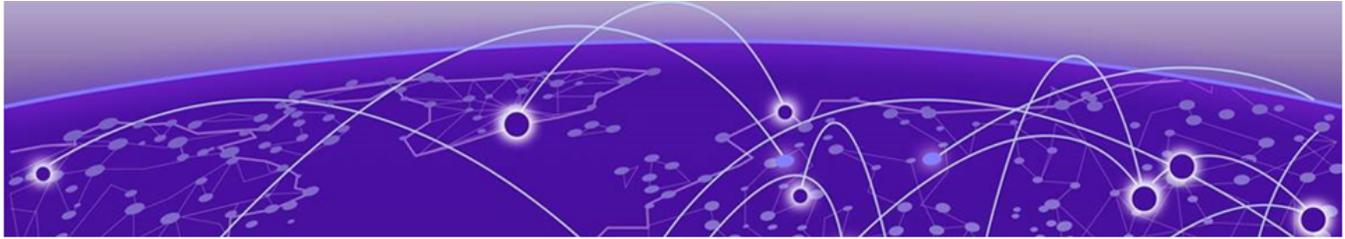
```

- **clear counters lacp:** Clears statistics for member ports of the dynamic LAGs that are configured on the device.

```

device# clear counters lacp
device# clear counters lacp interface port-channel 10-20
device#

```



Multi-Chassis Link Aggregation (MLAG)

[Overview](#) on page 22

[MLAG Limitations](#) on page 23

[MLAG Terminology](#) on page 24

[MLAG Control Plane](#) on page 24

[MLAG Resiliency](#) on page 26

[Using MLAG CLI commands](#) on page 33

[Configure MLAG Session on Peer Devices](#) on page 36

[MLAG Event Monitoring and Notification](#) on page 43

Overview

Multi-Chassis Link Aggregation (MLAG) is trunking that initiates at a single MLAG-unaware server or device and terminates at two MLAG-aware devices. MLAG allows the links to the two port channels (MLAG-aware devices) to appear to a downstream device as if they are coming from a single device on a single Link Aggregation (LAG) trunk interface or physical port.

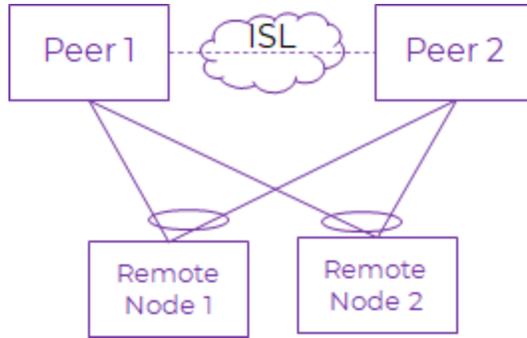
In a data center network environment, LAG trunks provide link level redundancy and increased capacity. However, they do not provide device-level redundancy. If the device connected to the LAG trunk fails, the entire trunk loses network connectivity.

With MLAG, member links of the LAG trunk are connected to two MLAG-aware devices. Logical Inter-Switch Link (ISL) configuration between the 2 MLAG devices enables data flow and control messages between them.

In this model, if one MLAG device fails, a data path remains through the other device.

Extreme ONE OS Layer 2 MLAG control plane protocol (MCP) synchronizes MAC and ARP data between the MLAG peers, for node resiliency and faster convergence.

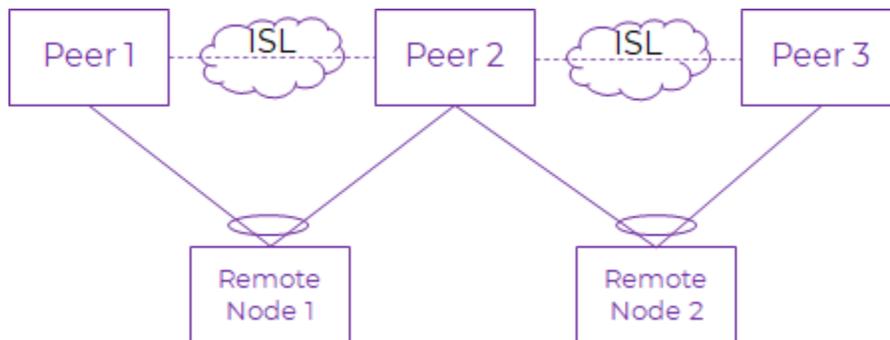
The data plane is established using a VxLAN tunnel between MLAG peers.



MLAG Limitations

Only a single MLAG peer device is allowed in this release.

The following topology, which contains multiple peers, is not supported in this release:



Other Limitations

- MLAG keep-alive source-interface as VE is not allowed.
- MLAG Primary keep-alive is supported only in the default-vrf.
- MLAG ID configuration as AUTO is not supported on Ethernet MLAG interface.
- Copper ports may remain operationally UP during a forward-hal crash.

Supported Features

MLAG only supports system fault generation for the following:

- Uncontrolled restart of the Fwd-HAL microservice
- Out of memory conditions

MLAG Terminology

MLAG	Multi-Chassis Link Aggregation
ISL	Inter-Service Link. This link is used to connect the two physical devices that form an MLAG. The ISL is the dedicated connection between the two MLAG peer devices. It is essential for the operation of MLAG and performs control plane communication, data traffic forwarding, and loop prevention.
VxLAN	Virtual Extensible LAN
MCP	MLAG Control Protocol
MLAG VLANs	VLANs that are shared by the MLAG peers. These VLANs are explicitly configured in the MLAG's Bridge-domain configuration.
Member PO	A port channel (PO) that is a member of an MLAG group. These are the ports that belong to the port channel that forms the MLAG link. These links connect a device to both of the MLAG peer devices. For example, a server might have two NICs, each connected to a different device, and these NICs are bundled into a port channel that is a member of the MLAG. This makes the server connection appear as a single, highly available, and high-bandwidth link.
Non-MLAG link	A standard network link that is not part of an MLAG configuration. In contrast to MLAG, a non-MLAG link is a simple connection between two devices or a standard Link Aggregation Group (LAG) that combines only multiple links into a single logical one between two devices. A non-MLAG link is any link that does not use this specific technology.
BD	Bridge Domain
VNI	Virtual Network Identifier
VE	Virtual Ethernet interface

MLAG Control Plane

Primary keepalive session based on gRPC connection using port 4012 is established between the nodes using the configured primary keepalive's peer IP and the IP of source interface. The primary keepalive session helps in the establishment of an initial connection. If BFD (Bidirectional Forwarding Detection) service is available, the session will be offloaded to BFD for further management, for example, monitoring.

Secondary keepalive session using TCP connection using port 4000 is established between the nodes using the configured secondary keepalive's IP and the IP of source interface if manually configured. By default, secondary keepalive establishes connection through management interface automatically.

Inter-Switch Link for the Extreme ONE OS SR Devices

The Inter-Switch Link (ISL) for the devices is a VxLAN tunnel created between the MLAG nodes. The destination IP address of the ISL tunnel is the MLAG node's Peer IP. The ISL source IP is configured on the source-interface given in the primary keepalive session.

The underlay interface carrying the traffic can be any physical port or port-channel Layer 3 interface between the MLAG peers. VEs are not supported. By default, all MLAG VLANs or bridge domains (BDs) are extended to the MLAG peer.

By default, VLAN-VNI mapping is automatically configured for the ISL VxLAN tunnel. Since a single VLAN-VNI mapping domain is supported, any change to this mapping under the overlay gateway changes the mapping for the ISL and temporarily affects its traffic.

MLAG Node Role Selection in AUTO Role

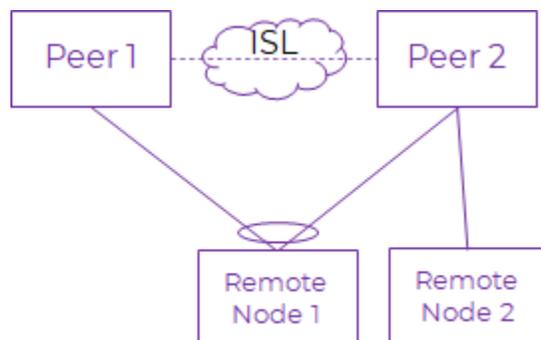
MLAG node role elected based on Primary Keepalive's Source IP in the AUTO role, otherwise Node with highest primary keepalive Source IP will be elected as PRIMARY and lowest will be BACKUP node. It is recommended to configure such that if one peer is configured as AUTO role, other node also must be configured as AUTO role.

MLAG Control Plane Protocol (MLAG CCP)

MLAG peers periodically send health-check (HC) messages. These messages are sent over an gRPC connection to the peer's 4012 Port number. Peers send health-check hellos every keepalive-interval ms, where keepalive-interval is a configurable parameter with a range 100 ms -1000000 ms, defaulting to 300 ms. Primary Keepalive must be configured for peer liveness detection. Peer will be declared DOWN if both Primary and Secondary keepalive status are DOWN.

In addition to keepalive messages, MCP also synchronizes MAC and ARP messages and exchanges system parameters, including management IP, configured bridge domains (BDs), and System MAC address.

The Primary Keepalive service establishes the initial connection between peers, which is then offloaded to BFD (Bidirectional Forwarding Detection) for ongoing monitoring. BFD utilizes an MLAG (Multi-Chassis Link Aggregation) profile to configure transmission intervals and detection multipliers. By default, BFD operates with a 300-millisecond interval and a detection multiplier of 3.



MLAG Resiliency

MLAG Resiliency enhances the fault tolerance of MLAG by detecting and responding to critical system failures such as out of memory condition or uncontrolled restart of FWD-HAL service of the primary MLAG device. MLAG resiliency feature ensures continuous network availability.

The primary objective is to prevent split brain mode during system faults and maintain acceptable convergence times.

Health Monitoring

The health monitoring logic is embedded in the monitoring service, which acts as a gRPC server. It listens for system fault events from various services, including memory and CPU utilization issues. When a fault is detected, it is published on the message bus, making it available to all subscribed services—including MLAG.

Use the following command to check the local node health:

```
curl 0:9004/show-peerdb

[admin@leaf]# curl 0:9004/show-peerdb

Dumping MLAG Config Data Structures:
*****

-----
Keepalive Interval      : ---
Keepalive Delay        : ---
Bringup Delay          : 30
Multiplier              : ---
Role                   : ---
Mac                    : ---
Mgmt IP                : 10.38.59.158 Idx: 22000001
System MAC             : 00:16:3e:54:e1:00
BringUpDelayTmrSt      : false
Local MaintenanceMode  : Disabled
Local Health           : Healthy
```

Use the following command to check the remote node health:

```
device# show mlag peer

Peer dut3
=====
Peer State           : UP
MCP State            : UP
Role                 : BACKUP
Elected MAC         : 02:00:22:33:44:55
Extend Bridge Count  : 46
Peer Exception       : Peer Under Unhealthy State
```

Split-Brain Handling

A split-brain scenario occurs when the primary keepalive link is down but the secondary keepalive remains active. In this case, the MLAG backup node typically shuts down all its client ports to prevent network inconsistencies. However, if the backup


```

Ethernet 0/5:2
Eth 0/5:3      9216      DOWN      DOWN      25G      0x10000a3
Ethernet 0/5:3
Eth 0/5:4      9216      DOWN      DOWN      25G      0x10000a4
Ethernet 0/5:4
Eth 0/6        9216      UP        DOWN (SYSTEM_FAULT) 100G     0x10000c0
Ethernet 0/6
Eth 0/7        9216      UP        DOWN (SYSTEM_FAULT) 100G     0x10000e0
Ethernet 0/7
Eth 0/8        9216      UP        DOWN (SYSTEM_FAULT) 100G     0x1000100
Ethernet 0/8
Eth 0/9        9216      UP        DOWN (SYSTEM_FAULT) 100G     0x1000120
Ethernet 0/9
Eth 0/10       9216      UP        DOWN (SYSTEM_FAULT) 100G     0x1000140
Ethernet 0/10
Eth 0/11       9216      UP        DOWN (SYSTEM_FAULT) 100G     0x1000160
Ethernet 0/11
:
:
:

Eth 0/31:1     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x10003e1
member port of po 78
Eth 0/31:2     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x10003e2
member port of po 78
Eth 0/31:3     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x10003e3
member port of po 78
Eth 0/31:4     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x10003e4
member port of po 78
Eth 0/32:1     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x1000401
member port of po 78
Eth 0/32:2     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x1000402
member port of po 78
Eth 0/32:3     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x1000403
member port of po 78
Eth 0/32:4     9216      UP        DOWN (SYSTEM_FAULT) 25G      0x1000404
member port of po 78
Po 18          9216      UP        DOWN      0G      0x4000012
Port-Channel 18
Po 28          9216      UP        DOWN      0G      0x400001c
Port-Channel 28
Po 78          9216      DOWN     DOWN     0G      0x400004e   Po b/w
9150-1, Leaf3, Leaf4
Po 101         9216      UP        DOWN     0G      0x4000065   Port-
Channel 101
Po 102         9216      UP        DOWN     0G      0x4000066   Port-
Channel 102
Tu ISL_10.7.8.7      UP        DOWN     0x3100001
Tunnel ISL_10.7.8.7
Ve 101         9216      UP        DOWN (NO_ACTIVE_MEMBERS) 0x5000065   VE
101

Device# show system health

-----
DEGRADED-COMPONENTS
-----

Component: Memory
Reason: insufficient memory available (< 953.7 MiB)
Time: 2025-02-03 15:04:05
    
```

When an OOM condition is detected on the device, MLAG MS communicates node unhealthy information to peer node using keepalive and system services.

When an OOM condition is cleared, MLAG MS starts the BringUp delay timer. Once the timer expires, the MLAG interfaces and uplink track interfaces are brought up, and MLAG MS sends a Node Health notification to the peer node once all interfaces are up.

Split-Brain Handling

A split-brain scenario occurs when the primary keepalive link is down but the secondary keepalive remains active. In this case, the MLAG backup node typically shuts down all its client ports to prevent network inconsistencies. However, if the backup node has already received a peer unhealthy notification, it will not shut down its client ports to avoid traffic blackholing.

Fault Flapping Prevention

- Monitors the number of flaps
- If flaps exceed 5, the fault will not be cleared

OOM Event Update

Updates specific key paths with OOM event information. For example, OOM event will be updated on following key path:

```
key /components/component [name=degraded-component]
{
  "name": "degraded-component",
  "subcomponents": {
    "subcomponent": [
      {
        "name": "Memory",
        "state": {
          "name": "Memory"
        }
      }
    ]
  }
}

key /components/component [name=Memory]
{
  "name": "Memory",
  "properties": {
    "property": [
      {
        "name": "reason",
        "state": {
          "configurable": false,
          "value": "insufficient memory available (less than 953.7 MiB)"
        }
      },
      {
        "name": "timestamp",
        "state": {
          "configurable": false,
          "value": "2025-02-03 06:59:36"
        }
      }
    ]
  }
}
```

```

    {
      "name": "count",
      "state": {
        "configurable": false,
        "value": "0"
      }
    }
  ],
  "state": {
    "name": "Memory",
    "type": "health"
  }
}

```

Show System Health CLI

Displays description of OOM event, including:

- Memory usage statistics
- Degraded components (Memory)
- Reason for degradation (insufficient memory available)

```

device# show system health
BIOS Vendor: SeaBIOS
BIOS Version: 1.13.0-lubuntu1.1
BIOS Date: 04/01/2014
Kernel Version: 5.10.210-yocto-standard
Kernel Arch: x86_64
CPU Procs: 6
OS: linux
Platform: alpine
Platform Version: 3.20.3
Memory Total: 7.63GB
Memory Free: 0.91GB
Memory Used: 6.48GB
Memory Used (%): 84.00%
CPU Model: Intel Xeon Processor (Cascadelake)
Cpu Load: 37
Primary Rootfs Disk Total: 16.37GB
Primary Rootfs Disk Free: 11.70GB
Primary Rootfs Disk Used: 3.82GB
Secondary Rootfs Disk Total: 17.00GB
System Uptime: 4m31s
RAM Caches(kB): 911252

RAM Free(kB): 227316
Ram Used(kB): 6842488
RAM Buffers(kB): 23052
Cpu SoftIRQ time: 0
Cpu System time: 3
Cpu IOWait time: 7
Cpu User time: 26
USRDATA Partition Used: 22.00%
APPDATA Partition Used: 8.00%
Secondary Rootfs Disk Used: 3.90GB
Disk Written(kB): 0
Secondary Rootfs Disk Free: 12.00GB
CONFIG Partition Used: 1.00%
IAH Partition Used: 1.00%
Disk Read(kB): 4184

-----
DEGRADED-COMPONENTS

```

```

-----
Component: Memory
Reason: insufficient memory available (less than 953.7 MiB)
Time: 2025-02-03 06:59:36

```

Uncontrolled Restart of FWD-HAL

When FwdHAL restarts, the MLAG service relies on health monitoring updates from the State Database (SDB) published by the monitoring service. If the monitoring service marks a node as unhealthy, MLAG receives this notification and promptly informs its peer node using the keepalive service. Similarly, when the system is declared healthy again, MLAG communicates this recovery to the peer, ensuring both nodes remain synchronized in their health status.

When FWD-HAL terminates ungracefully, the device reloads after a support save and graceful termination of services. To minimize traffic disruption:

1. Monitor-svc service detects crash: generates system fault and notifies services
2. Disables TX on QSFP interfaces: using host script to bring down front panel link
3. Copper ports not affected: only QSFP interfaces are disabled

For example, this event will be notified in the following keypath:

```

key /components/component[name=degraded-component]
{
  "name": "degraded-component",
  "subcomponents": {
    "subcomponent": [
      {
        "name": "Service",
        "state": {
          "name": "Service"
        }
      }
    ]
  }
}

```

```

key /components/component[name=Service]
{
  "name": "Service",
  "properties": {
    "property": [
      {
        "name": "reason",
        "state": {
          "configurable": false,
          "value": "service fwd-hal not healthy"
        }
      },
      {
        "name": "timestamp",
        "state": {
          "configurable": false,
          "value": "2025-02-03 07:06:27"
        }
      },
      {
        "name": "count",
        "state": {

```

```

    "configurable": false,
    "value": "0"
  }
}
],
},
"state": {
  "name": "Service",
  "type": "health"
}
}
}

```

System Fault Notification

- Updates specific key paths with fault information
- The **show system health** CLI displays reason for degradation (service fwd-hal not healthy)

```

device# show system health

-----
DEGRADED-COMPONENTS
-----
Component: Service
Reason: service fwd-hal not healthy
Time: 2025-02-03 07:06:27

```

Uncontrolled Restart of Multiple Services

When two or more services terminate ungracefully within a 10 mins interval, the device automatically reboots. To aid traffic convergence and minimize disruption, generate a system fault.

YANG Modules and CLI Commands

Yang Path

The monitor-svc service updates the SDB with system fault information in the following paths:

1. Degraded Component: lists all subcomponents in a degraded state - /components/component[name=degraded-component]
2. Subcomponent Details: provides reason, timestamp, and count for each subcomponent - /components/component[name=Memory]

Clearing Events

- Delete notification is sent on the subcomponent to clear the event
- /components/component[name=degraded-component] is updated with an empty subcomponent list

CLI Commands

For more information regarding MLAG CLI commands see the *Extreme ONE OS Switching Command Reference Guide*.

Show Command

When an Out-of-Memory (OOM) condition is detected:

- **Front panel Ethernet ports** become operationally down with reason code `SYSTEM_FAULT` (if admin UP)
- **show int brief** displays interface status, including OOM-affected ports
- **show system health** displays degraded components, including Memory, with the reason for degradation

Enhanced show mlag peer Command

- Displays peer node health state
- New leaf `/mlag/peers/peer[name=%v]/exception-state` updates peer exception information

The following example output shows Peer State (UP), Peer Exception (Peer Under Maintenance Mode, Unhealthy State):

```
device# show mlag peer
Peer dut3
=====
Peer State           : UP
MCP State            : UP
Role                  : BACKUP
Elected MAC         : 02:00:22:33:44:55
Extend Bridge Count  : 46
Peer Exception       : Peer Under Unhealthy State
```

RASLog Messages

- **MLAG Peer Unhealthy**
LogID: 17014 - "MLAG Peer 10.2.5.5 is Unhealthy"
- **MLAG Peer Healthy**
LogID: 17014 - "MLAG Peer 10.2.5.5 is Healthy"

Using MLAG CLI commands

This task uses the MLAG CLI commands to configure MLAG interfaces, keepalive behavior, and peer attributes, and then verify MLAG status and learned MAC addresses.

For details on syntax and command parameters, see the *Extreme ONE OS Switching Command Reference*.

1. Clear existing MLAG peer statistics so that new counters reflect only the current session.

```
device# clear counters mlag peer m12345
```

Replace *m12345* with the name of the MLAG peer whose counters you want to clear.

2. Configure MLAG interfaces and their identifiers.
 - a. In MLAG configuration mode, add an Ethernet interface and assign an MLAG ID.

```
device(config)# mlag
device(config-mlag)# interface ethernet 0/1 id 234
```

The ID value can be in the range 0–4294967295 or set to **auto**.

- b. Add one or more port-channel interfaces as MLAG interfaces.

```
device(config-mlag)# interface port-channel 10 id 1020
```

3. Set MLAG keepalive timers.

```
device(config)# mlag
device(config-mlag)# keepalive interval 1000 delay 100 multiplier 4
```

The keepalive command supports:

- interval (0–1000000 ms)
- delay (100–1000000 ms)
- multiplier (1–50)

Use the **no** form to restore default values.

4. Configure the global MLAG MAC address and device role.

- a. Specify the MLAG MAC address.

```
device(config)# mlag
device(config-mlag)# mac a001.a002.a003
```

Use 48-bit dotted notation (xxxx.xxxx.xxxx).

- b. Set the MLAG role for the device.

```
device(config-mlag)# role primary
```

The role can be **auto**, **primary**, or **backup**. With **auto**, the role is chosen based on keepalive behavior.

5. Configure the MLAG peer and extended bridge-domain behavior.

- a. Enter MLAG peer configuration mode and specify a peer name.

```
device(config)# mlag
device(config-mlag)# peer peer01
device(config-mlag-peer-peer01)#
```

- b. Extend a subset of bridge domains to the peer.

```
device(config-mlag-peer-peer01)# extend-bd subset 1,5,100-200
```

BDRANGE supports ranges such as 1–3, 5, 7–9 within 1–8192.

- c. Optionally, extend all bridge domains except those specified.

```
device(config-mlag-peer-peer01)# extend-bd except 1001,1301
```

6. Configure MLAG peer keepalive address type, destination, source interface, and keepalive type.

- a. Create or select a keepalive instance for the peer.

```
device(config-mlag-peer-peer01)# keepalive keepalive01
device(config-mlag-peer-keepalive-keepalive01)#
```

- b. Specify the address type for the keepalive destination.

```
device(config-mlag-peer-keepalive-keepalive01)# address-type manual
```

Use **auto** to have the system derive the management-based keepalive automatically or **manual** to specify explicit addresses.

- c. Configure the keepalive destination address.

```
device(config-mlag-peer-keepalive-keepalive01)# destination 12.1.1.1
```

The destination can be an IPv4 or IPv6 address.

- d. Select the source interface for the keepalive traffic.

```
device(config-mlag-peer-keepalive-keepalive01)# source-interface port-channel 10
```

Supported source interfaces include Ethernet, loopback, management, and port-channel.

- e. Set the keepalive type.

```
device(config-mlag-peer-keepalive-keepalive01)# type primary
```

The type can be **primary** or **secondary**. Typically, the primary keepalive uses the ISL path, and the secondary uses the management network.

7. Verify MLAG peer status, interfaces, MAC learning, and counters.

- a. Display MLAG counters for the peer and keepalive sessions.

```
device# show counters mlag peer

Peer peer01
=====
Transitions      : 2
No. of Restarts  : 2
Last clear       : 0000-00-00 00:00:00

Keepalive keepalive01
-----
Transitions      : 1
Receive         : 8 (BFD)
Transmit        : 8 (BFD)
Last clear       : 0000-00-00 00:00:00
device#
```

- b. Display MAC addresses learned via MLAG for a specific bridge domain.

```
device# show mac-address-table bridge-domain 100 mlag

Total number of Mac Entries: 2
Hardware Status Codes - #:Failed
Mac-Address          Type                Interface
-----
00:10:94:10:01:01    MLAG                ethernet 0/1
device#
```

- c. Check the overall MLAG configuration.

```
device# show mlag

Role           : PRIMARY
MAC            : 00:10:94:10:01:01
Bring up delay : 1000 seconds
Keepalive Delay : 100 ms
Tx Interval    : 1000 ms
Multiplier     : 4
No. of Peers   : 1
No. of Interfaces : 2
Uplink-track Interfaces : ethernet 0/1, port-channel 10

Peer Info:
=====
Peer           State
-----
peer01         UP

Interfaces Info:
=====
Id   Interface          Peer           Local/Remote State  Exceptions
-----
234  ethernet 0/1        peer01         UP
```

```

peer01                                     UP
Interfaces Info:
=====
Id      Interface                Peer                Local/Remote State  Exceptions
-----
1020    port-channel 10            peer01              UP
device#

```

- d. Check the MLAG interface summary.

```

device# show mlag interfaces

Interface Identifier: 234
  Interface: ethernet 0/1
Peer      Lcl/Rmt State  Rmt Cfg/Act Member  Up Count  Exceptions
-----
peer01    UP / UP        1 / 1      11
-----

Interface Identifier: 1020
  Interface: port-channel 10
Peer      Lcl/Rmt State  Rmt Cfg/Act Member  Up Count  Exceptions
-----
peer01    UP / UP        2 / 2      16
device#

```

- e. Display detailed information about MLAG peers and their keepalives.

```

device# show mlag peer

Peer peer01
=====
Peer State           : UP
MCP State            : UP
Role                  : PRIMARY
Elected MAC         : 50:0a:9c:97:90:00
Extend Bridge Count  : 1614
Peer Exception       : None

Keepalive keepalive01
-----
Destination Address: 12.1.1.1
State                : UP
Type                  : PRIMARY
Address Type         : MANUAL
Source Interface     : port-channel 10 ( 10.1.1.0 )
First Rx             : 0000-00-00 00:00:00
Last Rx              : 0000-00-00 00:00:00
device#

```

Configure MLAG Session on Peer Devices

You can configure MLAG sessions on peer devices.

Follow this procedure to configure MLAG session on peer devices at global configuration mode.

1. Enter the configure terminal command to access global configuration mode.

```
device# configure terminal
```

2. Run the **mlag** command to add an MLAG session at the global configuration level.

```

device# (config)# mlag
device# (config-mlag)#
peer                                MLAG peer mode
device# (config-mlag)# peer

```

```

NAME (String: 1-64 character) example. peer (peer1)
device# (config-mlag)# peer mlag-peer
device(config-mlag-peer-mlag-peer)#

```

3. Verify the MLAG session configuration on the peer devices.

```

#show running-config mlag
 mlag
  peer mlag-peer

#show running-state mlag
 mlag
  peer p1

```

4. Run the **uplink-track interface** command to configure uplink track ports between MLAG peers and external router.

```

uplink-track interface ( ethernet IFNAME | port-channel PONUMBER )
EX:
(config-mlag)# uplink-track interface
 ethernet      Ethernet
 port-channel  Port-channel

device(config-mlag)# uplink-track interface ethernet
 IFNAME Interface name in slot/port or slot/port:breakout format i.e slot/
port:<channel range>
device(config-mlag)# uplink-track interface ethernet 0/1:1

device(config-mlag)# uplink-track interface port-channel
 PORANGE Value from 1-255, Example: 1. Range Example: 1-3,5,7-9
device(config-mlag)# uplink-track interface port-channel 200

```

Uplink track ports are configured between MLAG peers and an external router. This command helps reduce convergence for reload cases by diverting traffic to alternate paths. After reload, uplink track interfaces are in the Oper down (MLAG_SHUT) state until the Bring Up delay timer expires.

In a split brain scenario, on a secondary MLAG peer, along with MLAG interfaces, uplink track interfaces are made Oper down (MLAG_SHUT).

5. Run the **clear counters mlag** command to clear the counters of a multichassis link aggregation (MLAG) peer.

```
clear counters mlag peer peer-name
```

The following example clears the counters on an MLAG peer named m12345:

```
device# clear counters mlag peer m12345
device#
```

6. Run the **extend-bd** command to extend the bridge domains across the multichassis link aggregation (MLAG) peer.

The following example configures the subset extend bridge domain mode with the range of bridge domains. In this example, the bridge domains with IDs 1,5,100-200 are extended to other MLAG peers:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# peer peer01
device(config-mlag-peer-peer01)# extend-bd subset 1,5,100-200
device(config-mlag-peer-peer01)#

```

To specify all the bridge domains except the configured bridge domains:

```
device(config-mlag-peer-mlag-peer)#extend-bd except
BDRANGE Value from 1-8192, Example: 1. Range Example: 1-3,5,7-9
```

7. Run the **interface (MLAG configuration)** command to configure a multichassis link aggregation (MLAG) interface.

The following example configures an Ethernet interface on port 1 in slot 0 and specifies 234 as the MLAG identifier number:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# interface ethernet 0/1 id 234
device#
```

The following is an example command of adding port channels:

```
device(config-mlag)# interface port-channel
PORANGE Value from 1-255, Example: 1.
device(config-mlag)# interface port-channel 1 id auto

device(config-mlag)# interface port-channel
PORANGE Value from 1-255, Example: 1.
device(config-mlag)# interface port-channel 10 id 1020
```

8. Run the **keepalive** command to specify the delay, interval, and multiplier settings for multichassis link aggregation (MLAG) keepalive transmissions.

The following example sets the interval to 1000 milliseconds, the delay to 100 milliseconds, and the multiplier to 4:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# keepalive interval 1000 delay 100 multiplier 4
device(config-mlag)#
```

9. Run the **keepalive (MLAG peer configuration)** command to specify the MLAG peer keepalive name for multichassis link aggregation group (MLAG) keepalive transmissions and enter MLAG peer keepalive configuration mode.

The following example configures an MLAG peer named peer01 and configure a keepalive named keepalive01:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# peer peer01
device(config-mlag-peer-peer01)# keepalive keepalive01
device(config-mlag-peer-keepalive-keepalive01)#
```

10. Run the **destination (MLAG peer configuration)** command to specify the IP address for the keepalive destination of the multichassis link aggregation (MLAG) peer.

The following example configures an MLAG peer named peer01 and also configures IPv4 address 12.1.1.1 as its keepalive destination:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# peer peer01
device(config-mlag-peer-peer01)# keepalive keepalive01
device(config-mlag-peer-keepalive-keepalive01)# destination 12.1.1.1
device(config-mlag-peer-keepalive-keepalive01)#
```

11. Run the **mac (MLAG configuration)** command to specify the multichassis link aggregation (MLAG) MAC address.

The following example configures the MLAG MAC address on the device:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# mac a001.a002.a003
device(config-mlag)#
```

12. Run the **bringup-delay** command to set the multichassis link aggregation (MLAG) bringup delay.

```
device# configure terminal
device(config)# mlag
device(config-mlag)# bringup-delay 100
device(config-mlag)#
```

13. Run the **role (MLAG configuration)** command to specify the type of role for multichassis link aggregation (MLAG).

The following example sets the MLAG role as primary:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# role primary
device(config-mlag)#
```

14. Run the **show counters mlag** command to display statistics for multichassis link aggregation (MLAG).

The following example displays statistics for the MLAG peer:

```
device# show counters mlag peer

Peer dut3
=====
Transitions      : 2
show counters mlag
No. of Restarts  : 2
Last clear       : 0000-00-00 00:00:00

Keepalive kp-primary
-----
Transitions : 1
Receive     : 8 (BFD)
Transmit    : 8 (BFD)
Last clear  : 0000-00-00 00:00:00

Keepalive kp-secondary
-----
Transitions : 1
Receive     : 199603
Transmit    : 199603device#
Last clear  : 0000-00-00 00:00:00
```

15. Run the **source-interface** command to specify the source interface for the multichassis link aggregation (MLAG) peer keepalive.

The following example configures an MLAG peer named peer01 and also configures an Ethernet port named 0/1 as its keepalive source interface:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# peer peer01
device(config-mlag-peer-peer01)# keepalive keepalive01
device(config-mlag-peer-keepalive-keepalive01)# source-interface ethernet 0/1
device(config-mlag-peer-keepalive-keepalive01)#
```

16. Run the **type (MLAG peer keepalive configuration)** command to specify the multichassis link aggregation (MLAG) peer keepalive type.

The following example configures an MLAG peer named peer01 and also configures it to use a primary keepalive type:

```
device# configure terminal
device(config)# mlag
device(config-mlag)# peer peer01
device(config-mlag-peer-peer01)# keepalive keepalive01
device(config-mlag-peer-keepalive-keepalive01)# type primary
device(config-mlag-peer-keepalive-keepalive01)#
```

17. Run the **show mlag interfaces** command to display details about the multichassis link aggregation (MLAG) interfaces

The following example displays information about all MLAG interfaces that are configured on the device:

```
DUT4# sho mlag interfaces
Interface Identifier: 1010
  Interface: ethernet 0/13:1
Peer□  Lcl/Rmt State Rmt Cfg/Act Member Up Count Exceptions
-----
dut3           UP / UP           1 / 1           1

Interface Identifier: 4294967295
  Interface: ethernet 0/17:1
Peer□  Lcl/Rmt State Rmt Cfg/Act Member Up Count Exceptions
-----
dut3           UP / UP           1 / 1           1

Interface Identifier: 1103
  Interface: port-channel 103
Peer□  Lcl/Rmt State Rmt Cfg/Act Member Up Count Exceptions
-----
dut3           UP / UP           3 / 3           1

Interface Identifier: 2000
  Interface: port-channel 104
Peer  Lcl/Rmt State Rmt Cfg/Act Member Up Count Exceptions
-----
dut3           UP / UP           1 / 1           1
```

18. Run the **show mlag peer bfd** command to display Bidirectional Forwarding Detection (BFD) details about the multichassis link aggregation (MLAG) primary keepalive state on the device.

The following example displays BFD details on the MLAG peer that is configured on the device:

```
device# show mlag peer bfd
Peer mlag-peer
=====
Keepalive one
-----
Profile          : default
Session ID       : 1
Session Status   : Up
Device
```

19. Run the **show mlag peer** command to display details about the multichassis link aggregation (MLAG) peers that are configured on the device.

The following example displays information about the MLAG peer that is configured on the device:

```
DUT3# sho mlag peer

Peer dut4
=====
Peer State           : UP
MCP State            : UP
Role                 : PRIMARY
Elected MAC         : 02:00:22:33:44:55
Extend Bridge Count  : 72
Peer Exception       : None
  Keepalive kp-primary
  -----
  Destination Address: 12.1.1.1
  State               : UP
  Type                : PRIMARY
  Address Type        : MANUAL
  Source Interface    : port-channel 20 ( 11.x.x.x )
  First Rx            : 2025-06-24 15:34:32
  Last Rx             : 2025-06-24 15:34:33
```

20. Run the **show mlag peer extended-bridges** command to display information about the MLAG peer extended bridges that are configured on the device.

```
device# show mlag peer extended-bridges
Peer dut3
=====
No. of Extend Bridge: 34
Extend Bridges :
1,100,200-202,205,209,215-216,300-302,400,402,500-506,1001,2001,2500-2501,3001-3002,300
5-3
006,4000-4003,8192
device#
```

21. Run the **show mlag** command to display information about the multichassis link aggregation (MLAG) configuration.

The following example displays information about the MLAG configuration on the device:

```
DUT4# show mlag
Role           : BACKUP
MAC            : none
Bring up delay : 100 seconds
Keepalive Delay : 1000 ms
Tx Interval    : 300 ms
Multiplier     : 3
Uplink Track Intf : [] ethernet 0/4,0/6
                port-channel 54,64
No. of Peers   : 1
No. of Interfaces : 17
Peer Info:
=====
Peer           State
-----
dut3           UP
Interfaces Info:
=====
Id             Interface           Peer           Local/Remote State  Exceptions
-----
1010          ethernet 0/13:1      dut3           UP / UP
```

```

4294967295  ethernet 0/17:1      dut3      UP / UP
1103       port-channel 103      dut3      UP / UP
2000       port-channel 104      dut3□     UP / UP
1181       port-channel 181      dut3      UP / UP
1182       port-channel 182      dut3□     UP / UP
183183     port-channel 183      dut3      UP / UP
184184     port-channel 184      dut3      UP / UP
1185       port-channel 185      dut3      UP / UP
186186     port-channel 186      dut3      UP / UP
1187       port-channel 187      dut3      UP / UP
1188       port-channel 188      dut3□     UP / UP
189        port-channel 189      dut3      UP / UP
1190       port-channel 190      dut3      UP / UP
191919     port-channel 191      dut3      UP / UP
1192       port-channel 192      dut3      UP / UP
1193       port-channel 193      dut3      UP / UP
DUT4# sho mlag peer
Peer dut3
=====
Peer State           : UP
MCP State            : UP
Role                 : BACKUP
Elected MAC         : 02:00:22:33:44:55
Extend Bridge Count  : 72
Peer Exception       : None
  Keepalive kp-primary
  -----
  Destination Address : 11.1.2.1
  State□              : UP
  Type                : PRIMARY
  Address Type        : MANUAL
  source Interface    : port-channel 20 ( 11.1.2.2 )
  First Rx□           : 2025-05-20 19:09:49
  Last Rx             : 2025-05-20 19:42:03
  Keepalive kp-secondary
  -----
  Destination Address : 10.x.x.x
  State               : UP
  Type                : SECONDARY
  Address Type        : AUTO
  Source Interface□   : management 0 ( 10.3x.x.x )
  First Rx            : 2025-05-20 19:09:51
  Last Rx             : 2025-05-20 20:13:28

```

22. Run the **show running-config mlag** command to display the MLAG configuration that is running currently on a device.

```

device# show running-config mlag
mlag
  role backup
  bringup-delay 100
  uplink-track interface□ ethernet 0/4,0/6,0/11:2
  uplink-track interface□ port-channel 12,54,64
  interface port-channel 183 id 183183
  interface ethernet 0/17:1 id 4294967295
  interface port-channel 190 id auto
  interface port-channel 182 id auto
  interface port-channel 193 id auto
  interface port-channel 191 id 191919
  interface ethernet 0/13:1 id 1010
  interface port-channel 188 id auto
  interface port-channel 181 id auto
  interface port-channel 184 id 184184
  interface port-channel 189 id 189
  interface port-channel 104 id 2000

```

```

interface port-channel 187 id auto
interface port-channel 192 id auto
interface port-channel 103 id auto
interface port-channel 185 id auto
interface port-channel 186 id 186186
peer dut3
  extend-bd except 1001,1301
  keepalive kp-primary
    type primary
    address-type manual
    destination 11.1.2.1
    source-interface port-channel 20
  !
  keepalive kp-secondary
    type secondary
  !
!
!
!

```

MLAG Event Monitoring and Notification

MLAG event monitoring and notification involves tracking the operational status, health, and specific events of an MLAG deployment to ensure network redundancy and stability. Extreme ONE OS Switching offers several ways to monitor these events and provide notifications:

- Command Line Interface (CLI): You can use specific commands (such as **show counters mlag peer**, **show mlag**, **show mac-address-table bridge-domain**, or **show mlag interfaces**) to check current status and statistics on the devices directly.
- Event Notifications/Alarms:
 - SNMP Traps: Events can be configured to send SNMP notifications (traps) to a central event monitor.
 - RASlog/Logging: Events are typically logged in the system log file and can be forwarded to a remote logging server.

Supported Notifications

The following traps are generated when the peer is Up or Down:

Table 4: Extreme ONE MLAG MIB

Trap Names and OIDs	Varbinds	Description
extremeMlagPeerDownTrap 1.3.6.1.4.1.1916.1.63.0.1	extremeMlagPeerAddrType extremeMlagPeerAddr	Generated when MLAG peer goes down extremeMlagPeerAddrType: peer IP address type extremeMlagPeerAddr: peer IP address
extremeMlagPeerUpTrap 1.3.6.1.4.1.1916.1.63.0.2	extremeMlagPeerAddrType extremeMlagPeerAddr	Generated when MLAG peer comes up

EXTREME ONE MLAG MIB Definitions

- MIB Module: Defines objects and notifications for MLAG
- Objects:
 - extremeMlagPeerAddrType: Specifies peer IP address type
 - extremeMlagPeerAddr: Specifies peer IP address
- Notifications:
 - extremeMlagPeerDownTrap: MLAG peer is down
 - extremeMlagPeerUpTrap: MLAG peer is up

Following is an sample of an Extreme ONE OS Switching MLAG MIB:

```
EXTREME-ONE-MLAG-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
        Integer32, Unsigned32, Counter32, Counter64
        FROM SNMPv2-SMI
        -- RFC 2578

    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF
        -- RFC 2580

    InetAddress, InetAddressType
        FROM INET-ADDRESS-MIB

    sysName
        FROM SNMPv2-MIB

    extremeAgent
        FROM EXTREME-BASE-MIB;

extremeOneMlagMIB MODULE-IDENTITY
    LAST-UPDATED "202503110000Z" -- 11 March 2025 00:00:00 GMT
    ORGANIZATION "Extreme Networks, Inc."
    CONTACT-INFO
        "Postal: Extreme Networks, Inc.
         6480 Via Del Oro
         San Jose, CA 95119 USA
         Phone: +1 408 579-2800
         E-mail: support@extremenetworks.com
         WWW: http://www.extremenetworks.com"
    DESCRIPTION
        "initial version"
    ::= { extremeAgent 63 }

-- Top-level components of this MIB module.

extremeMlagNotifications OBJECT IDENTIFIER ::= { extremeOneMlagMIB 0 }

extremeMlagPeerAddrType OBJECT-TYPE
    SYNTAX InetAddressType
    MAX-ACCESS accessible-for-notify
    STATUS current
    DESCRIPTION
        "This object specifies the IP address type of the peer."
    ::= { extremeOneMlagMIB 1 }

extremeMlagPeerAddr OBJECT-TYPE
    SYNTAX InetAddress
    MAX-ACCESS accessible-for-notify
    STATUS current
```

```

DESCRIPTION
    "This object specifies the peer IP address."
::= { extremeOneMlagMIB 2 }

extremeMlagPeerDownTrap NOTIFICATION-TYPE
    OBJECTS {
        extremeMlagPeerAddr, -- peer address
        extremeMlagPeerAddrType, -- peer address type
        sysName -- The local host name assigned for this switch
    }
    STATUS current
    DESCRIPTION
        "MLAG peer is down"
::= { extremeMlagNotifications 1 }

extremeMlagPeerUpTrap NOTIFICATION-TYPE
    OBJECTS {
        extremeMlagPeerAddrType, -- peer address type
        extremeMlagPeerAddr, -- peer address
        sysName -- The local host name assigned for this switch
    }
    STATUS current
    DESCRIPTION
        "MLAG peer is up"
::= { extremeMlagNotifications 2 }

END

```

YANG Module for MLAG Status Retrieval

Peer Status

- Retrieved from YANG leaf object: /mlag/peers/peer/state/peer-state
- Module: extreme-mlag
- Peer state values:
 - UP: peer is up
 - DOWN: peer is down
 - UNKNOWN: peer state is unknown

```

module: extreme-mlag
  +--rw mlag
    +--rw peers
      | +--rw peer*[name]
      |   +--rw name          -> ../config/name
      |   +--ro state
      |   | +--ro name?      string
      |   | +--ro peer-state? mlag-entity-state

```

Keepalive Status

- Retrieved from YANG leaf object: /mlag/peer/keepalives/keepalive/state/keepalive-state
- Module: extreme-mlag
- Keepalive state values:
 - UP: keepalive is up
 - DOWN: keepalive is down

- UNKNOWN: keepalive state is unknown

```

module: extreme-mlag
+--rw mlag
+--rw peers
| +--rw peer* [name]
|   +--rw keepalives
|     +--rw keepalive* [name]
|       +--rw name      -> ../config/name
|       +--ro state
|         +--ro name?          string
|         +--ro keepalive-state? mlag-entity-state

typedef mlag-entity-state {
  type enumeration {
    enum UP {
      description
        "The entity is perceived to be up by the system.";
    }
    enum DOWN {
      description
        "The entity is perceived to be down by the system.";
    }
    enum UNKNOWN {
      description
        "The current state of the entity is not known to the system.";
    }
  }
  description
    "Type for state of MLAG entities";
}

```

Displaying MLAG Addresses in the MAC Address Table for a Bridge Domain

You can use the **show mac-address-table bridge-domain ID mlag** command to displays MAC addresses learned from MLAG clients. This command differentiates the addresses learned locally and remotely from the MLAG peers.

The following example displays MLAG addresses in the MAC address table for bridge domain 100 on the device:

```

device# show mac-address-table bridge-domain 100 mlag

Total number of Mac Entries: 2
Hardware Status Codes - #:Failed
Mac-Address              Type              Interface
-----
00:10:94:10:01:01       MLAG              ethernet 0/1:1.10
device#

```

MLAG Event Log Messages

Peer Up/Down Events

Log messages indicate MLAG peer operational state changes (UP/DOWN). For example, MLAG Peer 7.7.7.2 is operationally UP or MLAG Peer 7.7.7.2 is operationally DOWN.

```
{ "Level": "info", "Service": "mlog", "LogID": 17011, "Time": "2025-01-13 06:25:09.969 UTC +0000", "Msg": "MLAG Peer 7.7.7.2 is operationally UP" }
{ "Level": "info", "Service": "mlog", "LogID": 17011, "Time": "2025-01-13 06:28:49.735 UTC +0000", "Msg": "MLAG Peer 7.7.7.2 is operationally DOWN" }
```

Keepalive Up/Down Events

Log messages indicate MLAG keepalive state changes (UP/DOWN) for primary and secondary keepalives. For example, MLAG Keepalives with Peer 7.7.7.2 Primary:DOWN Secondary:UP.

```
{ "Level": "info", "Service": "mlog", "LogID": 17009, "Time": "2025-01-13 06:28:49.224 UTC +0000", "Msg": "MLAG Keepalives with Peer 7.7.7.2 Primary:DOWN Secondary:UP" }
{ "Level": "info", "Service": "mlog", "LogID": 17009, "Time": "2025-01-13 06:28:49.735 UTC +0000", "Msg": "MLAG Keepalives with Peer 7.7.7.2 Primary:DOWN Secondary:DOWN" }
```

Peer Health Events

Log messages indicate MLAG peer health state changes (Healthy/Unhealthy). For example, MLAG Peer 10.2.5.5 is Unhealthy or MLAG Peer 10.2.5.5 is Healthy.

```
LogID:17014 Msg:MLAG Peer 10.2.5.5 is Unhealthy
LogID:17014 Msg:MLAG Peer 10.2.5.5 is Healthy
```

Maintenance Mode Events

Log messages indicate MLAG peer maintenance mode state changes (Enabled/Disabled). For example, Maintenance Mode is Enabled on MLAG Peer 10.38.59.158 or Maintenance Mode is Disabled on MLAG Peer 10.38.59.158.

```
LogID:17013 Msg:Maintenance Mode is Enabled on MLAG Peer 10.38.59.158
LogID:17013 Msg:Maintenance Mode is Disabled on MLAG Peer 10.38.59.158
```

RASLogs

The following table describes RASLog IDs:

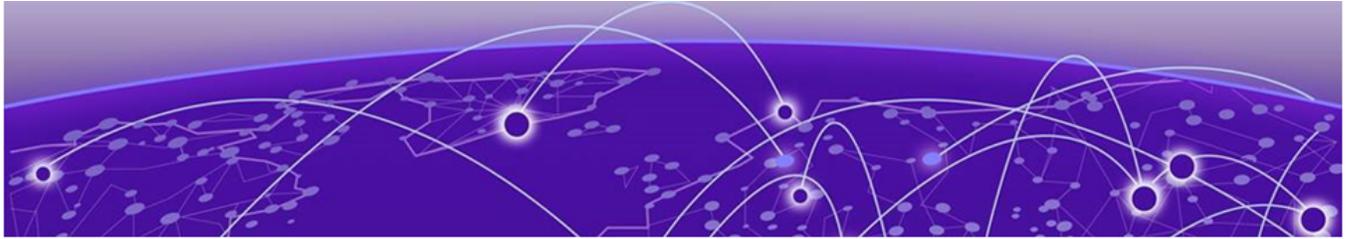
Raslog ID	Description	Sample log	Trigger
Raslog id	Description	Sample log	Trigger
17009	Mlag Keepalive UP event	MLAG Keepalives with Peer 1.1.1.1 Primary: UP Secondary: UP	MLAG is configured with primary and secondary keepalive.
17009	Mlag Keepalive DOWN event	MLAG Keepalives with Peer 1.1.1.1 Primary: DOWN Secondary: DOWN	Shut down the ISL interface
17011	Mlag Peer UP event	MLAG Peer 1.1.1.1 is operationally UP	This event will come if one of the keepalives is up. So, to simulate this event, bring up either of the keepalives.
17011	Mlag Peer DOWN event	MLAG Peer 1.1.1.1 is operationally DOWN	This event will come if both keepalives are down. So, to simulate this event, bring down both the keepalives.

Raslog ID	Description	Sample log	Trigger
17012	Mlag MCP UP event	MLAG Peer 10.2.5.5 MCP State operationally UP	MCP state depends on the primary keepalive. so, trigger is the same as primary keepalive.
17012	Mlag MCP DOWN event	MLAG Peer 10.2.5.5 MCP State operationally DOWN	MCP state depends on the primary keepalive. so, trigger is the same as primary keepalive.
17013	Maintenance Mode Enable event	Maintenance Mode is Enabled on MLAG Peer 10.38.59.158	Enable Maintenance Mode on peer MLAG device
17013	Maintenance Mode Disable event	Maintenance Mode is Disabled on MLAG Peer 10.38.59.158	Disable Maintenance Mode on peer MLAG device
17014	Unhealthy event	MLAG Peer 10.2.5.5 is Unhealthy	Trigger a fault in the peer node
17013	Healthy event	MLAG Peer 10.2.5.5 is Healthy	Clear the fault in the peer node



Note

MCP state depends on the primary keepalive state. MCP state will be UP if primary keepalive is UP, and DOWN if primary keepalive is DOWN.



Bridge Domains

[Bridge Domain Overview](#) on page 49

[Bridge Domain Configuration](#) on page 50

Bridge Domain Overview

Bridge domain is an infrastructure that supports the implementation of different switching technologies.

A bridge domain is a generic broadcast domain that is not tied to a specific transport technology. Bridge domains support a wide range of service endpoints including regular Layer 2 endpoints and Layer 2 endpoints over Layer 3 technologies.

A bridge domain determines the flooding domain, which is shared among all members attached to that bridge domain. This means that traffic flooded within the bridge domain will be propagated to all associated members.

Bridge Domain Limitations

Extreme 8730 Platform Limitations

When double-tagged traffic enters a single-tagged LIF and exits an untagged strict LIF, the inner VLAN remains unchanged, and only the outer VLAN is removed.

Bridge Domain Configuration

Use this topic to learn about configuring a bridge domain.



Note

- When a bridge domain member is configured using the "member" command in the CLI, it establishes the association only between the subinterface and the bridge domain. The subinterfaces must be created explicitly under the interfaces.
- Using the "no" form of the member command in the bridge domain will only remove the association of the subinterface with the BD; the subinterfaces themselves will remain intact under the interface.
- For information about commands and supported parameters to configure bridge domains, see *Extreme ONE OS SR Command Reference Guide*.

Configure Subinterface

You can configure subinterfaces under Ethernet and port channel interfaces to define specific packet-matching criteria. To manage packet flooding effectively, these subinterfaces must be linked to bridge domains, enabling proper packet forwarding and processing.

Follow the procedure to configure a subinterface.

1. Configure a tagged subinterface.

A tagged subinterface matches single-tagged packets with a specific VLAN ID and sends packets with the designated VLAN tag.

```
device(config)# interface ethernet 0/1
device(config-eth-0/1) # subinterface vlan 100
```

2. Configure untagged strict subinterface.



Note

Untagged strict subinterfaces are not supported on the Extreme 8730platform. All other platforms are supported.

An untagged strict subinterface matches only untagged packets received on the port and sends packets without a VLAN tag.

```
device(config)# interface ethernet 0/2
device(config-eth-0/1) # subinterface untagged
```

3. Configure untagged subinterface.



Note

This is supported only with VLAN-mode bridge domains (not with default-mode bridge domains).

An untagged subinterface matches untagged packets, associates them with a specified VLAN, and sends packets without a VLAN tag. This configuration also allows packets tagged with the VLAN ID associated with the VLAN-mode bridge domains.

```
device(config)# interface ethernet 0/3
device(config-eth-0/3)# subinterface untagged vlan 400
```

Configure Bridge Domain in VLAN Mode

VLAN Mode Bridge Domain: A bridge domain in VLAN mode determines VLAN switching behavior and is associated with a specific VLAN ID (1-4094). This mode supports the following key features:

- Tagged and untagged configurations
- Member ports added to the bridge domain create a subinterface linked to the bridge domain

Untagged Member Behavior: An untagged member in a VLAN mode bridge domain accepts both:

- Untagged packets
- Tagged packets with the configured VLAN ID

Follow the procedure to configure the bridge domain in VLAN mode.

1. Configure a bridge domain with VLAN mode BD.

```
bridge-domain <ID> [mode vlan]

DUT(config)# bridge-domain
(1-8192) Bridge domain Id(1-8192)
DUT(config)# bridge-domain 100 mode
vlan          Vlan mode
```

2. Configure VLAN-ID if the BD is a VLAN mode BD.

```
device(config)# bridge-domain 100 mode vlan
device(config-vlan-bd-100)# vlan-id
<1-4094> Vlan ID range <1-4094>
```

3. Configure the tagged members.

Add physical ports or port channels as tagged members to a VLAN mode bridge domain.

```
device(config)# bridge-domain 100 mode vlan
device(config-bd-100)# vlan-id 100
device(config-bd-100)# member ethernet 0/1
```

4. Configure the untagged members.

Add physical ports or port channels as untagged members, accepting both untagged packets and tagged traffic from the VLAN.

```
device(config)# bridge-domain 100 mode vlan
device(config-bd-100)# vlan-id 100
device(config-bd-100)# member ethernet 0/1 untagged
```

5. Configure the untagged strict members.

Add physical ports or port channels as untagged strict members, which only accept untagged packets. Requires creating an untagged strict subinterface and adding it to the bridge domain.

```
device(config)# interface ethernet 0/2
device(config-eth-0/1)# subinterface untagged

device(config)# bridge-domain 100 mode vlan
device(config-bd-100)# vlan-id 100
device(config-bd-100)# member ethernet 0/2 untagged
```

6. Configure VXLAN tunnel members.

Include VXLAN tunnels in a bridge domain for extended network connectivity.

```
device(config)# bridge-domain 100 mode vlan
device(config-bd-100)# member tunnel vxlan-tunnel-1
```

Configure Bridge Domain in Default Mode

Bridge Domain Default Mode: In default mode, a bridge domain enables extended bridging functionality, supporting various subinterface configurations:

- Single-tagged
- Double-tagged
- Untagged strict subinterfaces

Untagged Traffic Handling: An untagged member within the default mode bridge domain exclusively accepts untagged traffic via an untagged strict subinterface.

Follow the procedure to configure a bridge domain in default mode.

1. Run the following commands

```
device(config)# bridge-domain 100
device(config-bd-100)# member ethernet 0/1 vlan 100
device(config-bd-100)# member ethernet 0/1 vlan 200 inner-vlan 300
device(config-bd-100)# member ethernet 0/1 untagged
device(config-bd-100)# member tunnel vxlan-tunnel-1
```

2. Run the **member (tagged)** command to create a tagged subinterface and assign the specified bridge domain membership.

The following example creates a tagged subinterface and assigns the specified bridge domain membership:

```
device# configure terminal
device(config)# bridge-domain 200 mode vlan
device(config-vlan-bd-200)# vlan-id 200
device(config-vlan-bd-200)# member ethernet 0/2
device(config-vlan-bd-200)# member port-channel 1
device(config-vlan-bd-200)#
device# show running-config bridge-domain 200
bridge-domain 200 mode vlan
```

```
vlan-id 200
member ethernet 0/2
member port-channel 1
device#
```

3. Run the **member (untagged)** command to create an untagged subinterface and assign the specified bridge domain untagged membership:

The following example creates a tagged subinterface and assigns the specified bridge domain membership:

```
device(config)# bridge-domain 200 mode vlan
device(config-vlan-bd-200)# member ethernet 0/2 untagged
device# show running-config
bridge-domain 200 mode vlan
vlan-id 200
member ethernet 0/2 untagged
device(config)#
```

4. (Optional) Create a VNI domain, which defines the VNI name space used by tunnels.

```
device(config-vlan-bd-200)# vni-domain

device(config-vlan-bd-200)# vni-domain mydomain1 vni
<1-16777215> Vni value for bridge domain
```

The following example configures the VNI mapping bridge domain. This command sets the VNI ID to be used for this bridge domain when going over the VXlan tunnels (which are associated to the VNI domain named mydomain1):

```
device# configure terminal
device(config)# bridge-domain 200
device(config-bd-200)# vni-domain mydomain1 vni 64000
device(config-bd-200)# do show running-config bridge-domain 200
bridge-domain 200 mode vlan
vlan-id 200
member ethernet 0/21
vni-domain mydomain1 vni 64000
device(config-bd-200)#
```

Configure a Static MAC Address

Perform the following steps to configure static MAC address entries in a bridge domain on the device.

1. Run the following commands to configure static MAC address entries.

```
device# configure terminal
device(config)# bridge-domain 1
device(config-bd-1)# static-mac-address 40:88:2f:f9:c0:03 ethernet 0/1 vlan 1
device(config-bd-1)# static-mac-address f0:64:26:f5:c8:03 ethernet 0/2 vlan 401
device(config-bd-1)# static-mac-address f0:64:26:f5:c8:03 port-channel 2 untagged
device(config-bd-1)# static-mac-address f0:64:26:f5:c8:05 port-channel 2 vlan 10
device(config-bd-1)# static-mac-address 02:e0:52:11:11:11 ethernet 0/2 untagged
device(config-bd-1)# static-mac-address 02:e0:52:11:11:11 ethernet 0/4 vlan 10
device(config-bd-1)#
```

2. Run the following command to verify the bridge domain configuration on the device.

```
device# show running-config bridge-domain

bridge-domain 10 mode vlan
vlan-id 1
static-mac-address 40:88:2f:f9:c0:03 ethernet 0/1 untagged
```

```

bridge-domain 1
 static-mac-address 40:88:2f:f9:c0:03 ethernet 0/1 vlan 1
 static-mac-address f0:64:26:f5:c8:03 ethernet 0/2 vlan 401
 static-mac-address f0:64:26:f5:c8:03 port-channel 2 untagged
 static-mac-address f0:64:26:f5:c8:05 port-channel 2 vlan 10
 static-mac-address 02:e0:52:11:11:11 ethernet 0/2 untagged
 static-mac-address 02:e0:52:11:11:11 ethernet 0/4 vlan 10
device#

```

3. Run the following command to show the MAC address table for all bridge domains on a device:

```

device# show mac-address-table bridge-domain all

Bridge-Domain:1
-----
Total number of Mac Entries: 1
Hardware Status Codes - #:Failed
Mac-Address          Type          Interface
-----
00:10:00:00:00:01    Static        ethernet 0/3

Bridge-Domain:10
-----
Total number of Mac Entries: 1
Hardware Status Codes - #:Failed
Mac-Address          Type          Interface
-----
00:10:00:00:00:02    Static        ethernet 0/4.10

vm1# show mac-address-table bridge-domain 10 static
Total number of Mac Entries: 1
Hardware Status Codes - #:Failed
Mac-Address          Type          Interface
-----
00:10:00:00:00:02    Static        ethernet 0/4.10

vm1# show mac-address-table bridge-domain 10 00:10:00:00:00:02
'*' denotes best route-source
Mac-Address          Type          Interface          Last
Change              Seq No      Hardware Status
-----
*00:10:00:00:00:02    Static        ethernet 0/4.10
48s                  0
device#

```

Configure MAC Learning

MAC learning is always enabled on the bridge domains.

1. Run the following command to view the MAC learning information:

```

device# show mac-address-table bridge-domain all

Bridge-Domain:100
-----
Total number of Mac Entries: 6
Hardware Status Codes - #:Failed
Mac-Address          Type          Interface
-----
00:10:00:00:00:00    Dynamic        ethernet 0/1:1.100
00:10:00:00:00:06    Dynamic        ethernet 0/1:1.102

```

```
00:10:94:00:00:02      Dynamic      ethernet 0/1:1.100
device#
```

2. Run the following commands to clear the MAC address table:

```
device# clear mac-address-table bridge-domain ID static
device# clear mac-address-table bridge-domain ID dynamic
device# clear mac-address-table bridge-domain ID mlag
device# clear mac-address-table bridge-domain ID evpn
device# clear mac-address-table bridge-domain ID X:X:X:X:X
device#
```

The following is an example:

```
device# clear mac-address-table bridge-domain 100 static
device# clear mac-address-table bridge-domain 100 dynamic
device# clear mac-address-table bridge-domain 100 mlag
device# clear mac-address-table bridge-domain 100 evpn
device# clear mac-address-table bridge-domain 100 01:01:01:01:01:01
device#
```

3. Run the following command to display bridge domain information.

- To display the Bridge domain information:

```
device# show bridge-domain all

*untag-s --accepts untagged only *untag --accepts untagged+tagged

Bridge Domain 1      Mode L2VSI_VLAN Vlan 100
Total number of member ports 2
If Name   Vlan   Inner Vlan Admin Status Oper Status
=====  =====
Eth 0/1:1 --    --      UP        UP
Po 100   untag-s --      UP        UP

Bridge Domain 100   Mode L2VSI_P2MP
Vni Domain base Vni 64000
Total number of member ports 1
Member ports:
If Name   Vlan   Inner Vlan Admin Status Oper Status
=====  =====
Eth 0/7   100                UP        DOWN
device#
```

- To display the mac-address-table for a specific bridge domain.

```
device# show mac-address-table bridge-domain 100

Total number of Mac Entries: 3
Hardware Status Codes - #:Failed
Mac-Address      Type      Interface
-----
00:10:94:00:00:02 Dynamic  ethernet 0/13.100
00:10:94:00:00:10 Static   ethernet 0/13.100
00:16:3e:1e:b6:03 Local    Local
device#
```

- To display the mac-address-table for a specific bridge domain and specific source.

```
device# show mac-address-table bridge-domain 100 dynamic

Total number of Mac Entries: 3
Hardware Status Codes - #:Failed
Mac-Address      Type      Interface
-----
00:10:94:00:00:02 Dynamic  ethernet 0/13.100
device#
```

- To display details of ethernet interface or range of interfaces 1/1-2,2/1-2,3/2:1-4:

```
device# show interface ethernet 0/1 subinterface

Interface ethernet 0/1
Vlan      Inner Vlan Admin Status Oper Status If Mode
=====
untag(100) -          UP      DOWN   IF_MODE_L2
100       200          UP      DOWN   IF_MODE_L2
device#
```

- To display details of port-channels:

```
device# show interface port-channel 103 subinterface

Vlan      Inner Vlan Admin Status Oper Status If Mode
=====
untag(1000) -          UP      UP     IF_MODE_L2
1         0          UP      UP     IF_MODE_L2
2         0          UP      UP     IF_MODE_L2
device#
```

- To display global aging time for mac-address-table:

```
device# show system mac-address-table aging-time

Mac Aging Time: 1500
device#
```

Configure MAC Address Aging

MAC addresses that are dynamically learned are stored in MAC address table. The MAC address aging feature provides a mechanism to flush out the dynamic MAC addresses that remain inactive for a specified period.

The aging time of dynamic MAC address entries can be configured using the **mac-address-table aging-time** command. You can disable the MAC address aging by specifying the aging time as 0 (zero). The MAC aging time can be configured to a value from 60 through 38400 seconds. By default, the aging time of dynamic MAC address entries is 1800 seconds. The configured MAC aging time is applied to all MAC addresses in the system.

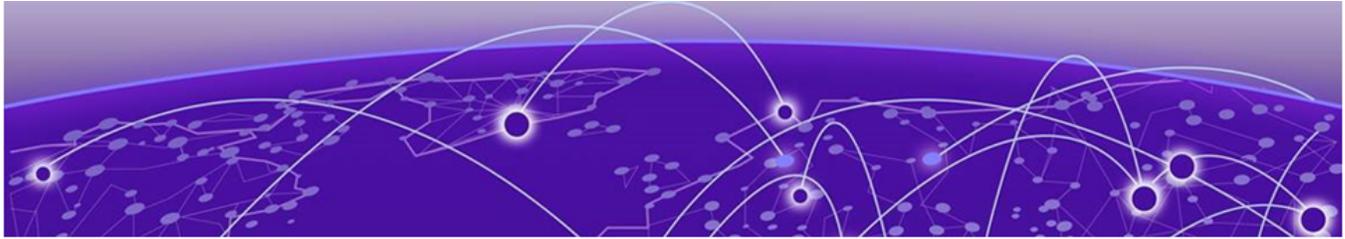


Note

The MAC address aging configuration per bridge-domain is not supported.

The following is an example configuration of MAC address aging (MAC forwarding table aging time):

```
device# configure terminal
device(config)# system
device(config-system)# global
device(config-system-global)# mac-address-table aging-time 1000
device(config-system-global)
```



MAC Movement Detection and Resolution

[MAC Movement Overview](#) on page 57

[MAC Movement Detection](#) on page 58

[MAC Movement Resolution](#) on page 58

[MAC Movement Detection and Resolution Commands](#) on page 60

MAC movement detection and resolution is a mechanism to prevent loop detection in networks. This feature fixes network malfunctions by detecting frequent MAC learns between different logical interfaces (also called detection of MAC mobility), then detecting the network loop that causes the mac mobility, and then shutting down the necessary ports or logging them so that you can take action manually.

- Repeated MAC movement detection and resolution works for switch ports.
- Loop detection using MAC movement detection is mutually exclusive of STP and ELD protocol operations.
- MAC movement detection is not considered for a port where port security is enabled and a restrict violation action is triggered.

MAC Movement Overview

A MAC address is defined as moved when the same MAC address is received on a different interface in the same VLAN. In MLAG, MAC movement is allowed on both local and remote nodes. However, a high MAC move rate can indicate a loop in the network or an issue with the server side interface, resulting in flapping. A high rate requires the control plane to process an extremely high rate of MAC learning events and can exhaust the control plane resources.

MAC Move Definitions

- **Rapid MAC movement:** A MAC that moves across multiple logical interfaces (LIF), ports, and VLANs is tracked for each second it moves. If the number of moves crosses the defined threshold, then it is treated as a MAC move violation.
- **Slow MAC movement:** In the first second that MAC movement is detected, the movement is monitored. In that first second, if the number of moves does not cross the threshold, then the MAC is tracked for a maximum of 10 seconds. If after 10 seconds the total number of moves is within the threshold, then no action occurs. If it exceeds the threshold limit, then an action is triggered.

MAC Movement Detection

This feature uses RAS traces to log MAC movement rules violations.

Once MAC movement detection is enabled and configured, the MAC move detection mechanism keeps track of all the movements of each MAC address between all the interfaces that they are learned on. If the number of MAC moves in one second exceeds the user defined threshold limit, then three values (old LIF, current LIF, and number of MAC moves) are recorded.

This list of recorded values is parsed automatically. If a port has a high number of LIFs with MAC moves, then those LIFs are recorded to the RASlog (by default) or are shut down (if this action is configured).

For example, MAC A moves between port 1, VLAN 10 and port 2, VLAN 10. MAC B moves between port 1, VLAN 20 and port 3, VLAN 20. MAC C moves between port 4, VLAN 10 and port 5, VLAN 10. All MACs have crossed the user defined threshold. The list looks like this:

- MAC A: port 1,VLAN 10; and port 2, VLAN 10
- MAC B: port 1, VLAN 20; and port 3, VLAN 20
- MAC C: port 4, VLAN 10; and port 5, VLAN 10

MAC Movement Resolution

Based on the example above, Port 1 is selected for the MAC movement action (**shutdown** | **raslog**) because it has a higher number of LIFs in the list. The LIFs port 1,VLAN 10 and port 1, VLAN 20 are either shut down with the actions logged, or the movement details are logged (without a shutdown).

The default MAC movement action is **raslog**. All MAC movement information is saved to the RASlog. When the auto-recovery time limit expires, the LIF becomes operational, and the tracking process starts again.



Note

Logical interfaces (LIFs) that are part of a tunnel, an Inter-Switch link (ISL), or a pseudowire (PW) are ignored and are not subject to any resolution action.

Automatic recovery is enabled by default. The LIF remains shut down for 5 minutes by default. The range is 3 through 30 minutes. The automatic recovery mode can be disabled from the command line.

Shutdown can be disabled from the command line, and logging enabled as the only action when the threshold is exceeded. The MAC movement details appear in the RASlog. For example:

```
device# 2025-06-03 21:45:18.6698 uftm[7]: Level:warn LogID:0 Topic:4 Msg:MAC-MOVE
DETECTION: MAC 00:11:22:33:44:55/48 exceeded the number of moves between ports!
Interface: port-channel 101.4088 Bridge Domain: 1
2025-06-03 21:45:20.6714 uftm[7]: Level:warn LogID:0 Topic:4 Msg:MAC-MOVE DETECTION: MAC
00:11:22:33:44:55/48 exceeded the number of moves between ports! Interface: port-channel
101.4088 Bridge Domain: 1
2025-06-03 21:45:35.6821 uftm[7]: Level:warn LogID:0 Topic:4 Msg:MAC-MOVE DETECTION: MAC
```

```
00:11:22:33:44:55/48 exceeded the number of moves between ports! Interface: port-channel
102.409610 Bridge Domain: 1
```

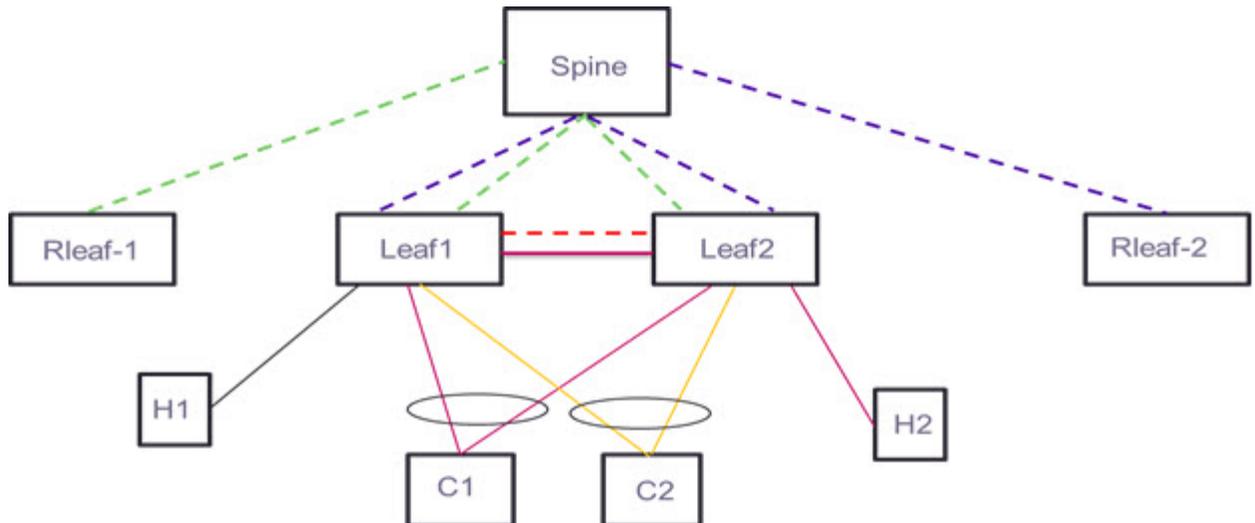


Figure 1: MAC movement detection scenario

MLAG Use Case: MAC loop between MLAG Interfaces

If MAC movement between C1 and C2 crosses the threshold MAC move limit, the LIF shutdown action is taken on the cluster node with the higher IP address. For example: Leaf1 has the higher cluster IP configured. Leaf1 takes the decision to shut down the LIF connecting to C1 and communicates to Leaf2 to shut down the corresponding LIF connecting to C1.

MLAG Use Case: MAC loop between MLAG Interface and Non-MLAG interface

If MAC movement between C1 and H1 crosses the threshold MAC move limit, the LIF shutdown action is taken on the MLAG node with the higher IP address. For example, Leaf1 has the higher cluster IP configured. Leaf1 takes the decision to shut down the LIF on C1 and communicates to Leaf2 to shut down the corresponding LIF on C1. If the MAC movement occurs between the MLAG Interface and Non-MLAG interface, the action is always taken on the MLAG Interface.

MLAG Use Case: MAC loop between Non-MLAG interfaces

If MAC movement between H1 and H2 crosses the threshold MAC move limit, the action is taken by each leaf independently. Leaf1 and Leaf2 will run MAC move detection separately.



Note

In an IP fabric configuration, this feature must be enabled on all leaf nodes.



Note

This feature is only supported on bridge domains.

:

MAC Movement Detection and Resolution Commands

The following commands support MAC movement detection and resolution. Run these command under the Global configuration (config mode > system> global) mode.

Table 5: MAC movement detection and resolution commands

Command	Details
12 mac-move-detection enable	Enables MAC movement detection. This feature is enabled by default. Use the no form of this command to disable the feature.
12 mac-move-detection threshold <i>moves</i>	Event count (maximum number of moves) after which the port is shut down or a RASlog is triggered. Use an integer from 3 to 500. The default is 20.
12 mac-move-detection action shutdown	Shuts down the port/interface when MAC movement exceeds the specified threshold. This is in addition to the generation of a RASlog entry (which always occurs). Shutdown is enabled by default when 12 mac-move-detection enable is enabled; use the no form of this command to disable shutdown. Details about every shutdown appear in the RASlog.
12 mac-move-detection auto-recovery-time <i>minutes</i>	Time (in seconds) until the port state is reset to up. Use an integer from 0 to 1800. 0 disables auto-recovery. The default is 300 (5 minutes). Details about every recovery appear in the RASlog.
12 mac-move-detection interval <i>seconds</i>	Interval (in seconds) after which the MAC move events are deleted. The default is 10 seconds. The range is 10 through 120 seconds.
clear 12 mac-move-detection shut-list [interface { ethernet <i>interface-name</i> port-channel <i>port</i> ve <i>interface-name</i> loopback <i>interface-name</i> }]	Clears entries from the shutdown list and sets the port state to up for all ports in a list or for a specified port.

Table 5: MAC movement detection and resolution commands (continued)

Command	Details
show bridge-domain all	<p>LIF output indicates when the LIF is down because of MAC movement. For example:</p> <pre>L1B-J2# show br 105 *untag-s --accepts untagged only *untag --accepts untagged+tagged Bridge Domain 105 Mode L2VSI_VLAN Vlan 105 Total number of member ports 3 If Name Vlan Inner Vlan Admin Status Oper Status ===== ----- Tu ISL_10.20.20.7 - - UP UP Po 101 -- DOWN - UP (MAC MOVE_SHUT) Po 201 -- UP - UP UP L1B-J2# show br 115 *untag-s --accepts untagged only *untag --accepts untagged+tagged Bridge Domain 115 Mode L2VSI_VLAN Vlan 115 Total number of member ports 3 If Name Vlan Inner Vlan Admin Status Oper Status ===== ----- Po 102 -- DOWN - UP (MAC MOVE_SHUT) Po 202 -- UP - UP UP Tu ISL_10.20.20.7 - - UP UP</pre>
show l2 mac-move-detection	<p>Displays the MAC move settings. For example:</p> <pre>device# show l2 mac-move-detection MAC-move detection is enabled and configured: Tracking interval: 17 seconds Threshold for triggering action: 10 moves Action taken when threshold is reached: shutdown Auto-recovery is done after 60 seconds. device#</pre>

Table 5: MAC movement detection and resolution commands (continued)

Command	Details
show l2 mac-move-detection shut-list	Displays the MAC move detection port shutdown list for a device. For example: <pre> device# show l2 mac-move-detection shutlist MAC-move detection shut port list Interface BD ID Time remaining (s) ----- ----- ethernet 0/1:3.100 500 2 ethernet 0/1:2.100 500 34 device# </pre>
show interface ethernet <i>interface-name</i> subinterface	Displays when the LIF/port subinterface is down because of MAC movement.

Table 5: MAC movement detection and resolution commands (continued)

Command	Details
<pre>system internal service uftm command mac-move-detection [args arguments]</pre>	<p>Displays the internal state of MAC move detection. It lists:</p> <ul style="list-style-type: none"> • The MAC move globals. • The current window index that is used to capture MAC moves. • The event map, which is the list of pairs between LIFs and number of moves per second for each bridge domain-MAC pair. • The detailed shut list, which is the same as shown by the show 12 mac-move-detection shut-list command but also adds the LIF information. <p>For example:</p> <pre>device# system internal service uftm command mac-move-detection MAC-move detection debug information Dumping macMove globals: enabled = true interval = 60 threshold = 18 action = shutdown shutInterval = 65 Current macMoveData.windowIndex = 52 Dumping macMoveData.eventMap: BD: 1, MAC: 00:a9:09:11:0a:01/48, moves: 2 = {0, []}, {0, []}, {2, [0x2000036, 0x3100002,]}, {0, []}, {0, []}, {0, []}, {0, []}, {0, []}, Dumping macMoveData.shutPortList: Interface LIF BD ID Time remaining (s) ----- ----- ethernet 0/1:1.30 0x200002c 1 63 device#</pre>
<pre>show system internal cdb path /system/global/12/mac- move-detection/config</pre>	<p>Displays the internal state database (SDB) records or the timestamp, user, and IP address of the last update for the /system/global/12/mac-move-detection/config data path.</p>

Table 5: MAC movement detection and resolution commands (continued)

Command	Details
<pre>show system internal sdb path /system/global/l2/mac- move-detection/state</pre>	<p>Displays the internal configuration database (CDB) records or the timestamp, user, and IP address of the last update for the /system/global/l2/mac-move-detection/state data path.</p>
<pre>show running-config system global system global l2 mac-move-detection enable l2 mac-move-detection action shutdown</pre>	<p>Displays the MAC movement detection feature is enabled (by default). Shuts down the port/interface when MAC movement exceeds the specified threshold. This is in addition to the generation of a RASlog entry (which always occurs). Shutdown is enabled by default when l2 mac-move-detection enable is enabled; use the no form of this command to disable shutdown. Details about every shutdown appear in the RASlog.</p>

For details about these commands, see the *Extreme ONE OS Switching Command Reference*.