



---

## Junos<sup>®</sup> OS

# Layer 2 Bridging, Address Learning, and Forwarding Feature Guide for Routing Devices

Release  
15.1



---

Modified: 2015-05-27

Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, California 94089  
USA  
408-745-2000  
[www.juniper.net](http://www.juniper.net)

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

*Junos<sup>®</sup> OS Layer 2 Bridging, Address Learning, and Forwarding Feature Guide for Routing Devices*

15.1

Copyright © 2015, Juniper Networks, Inc.  
All rights reserved.

The information in this document is current as of the date on the title page.

#### YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

#### END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# Table of Contents

	About the Documentation . . . . .	xi
	Documentation and Release Notes . . . . .	xi
	Supported Platforms . . . . .	xi
	Using the Examples in This Manual . . . . .	xi
	Merging a Full Example . . . . .	xii
	Merging a Snippet . . . . .	xii
	Documentation Conventions . . . . .	xiii
	Documentation Feedback . . . . .	xv
	Requesting Technical Support . . . . .	xv
	Self-Help Online Tools and Resources . . . . .	xv
	Opening a Case with JTAC . . . . .	xvi
<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Layer 2 Bridging . . . . .</b>	<b>3</b>
	Layer 2 Bridge Domains Overview . . . . .	3
	Layer 2 Virtual Switches Overview . . . . .	4
	Layer 2 Learning and Forwarding for Bridge Domains Overview . . . . .	5
	Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports . . . . .	5
	Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances . . . . .	6
<b>Chapter 2</b>	<b>Layer 2 Address Learning and Forwarding . . . . .</b>	<b>7</b>
	Layer 2 Learning and Forwarding Overview . . . . .	7
<b>Part 2</b>	<b>Configuration</b>	
<b>Chapter 3</b>	<b>Layer 2 Bridging and Layer 3 IP Routing . . . . .</b>	<b>11</b>
	Configuring a Bridge Domain . . . . .	11
	Example: Configuring Basic Layer 2 Switching on MX Series . . . . .	12
	Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances . . . . .	19
	Configuring Integrated Routing and Bridging for Bridge Domains . . . . .	24
	Configuring Bridge Domains as Switches for Layer 2 Trunk Ports . . . . .	26
<b>Chapter 4</b>	<b>Layer 2 Virtual Switches . . . . .</b>	<b>29</b>
	Configuring Integrated Routing and Bridging for a Bridge Domain in a Layer 2 Virtual Switch . . . . .	29
	Configuring VPLS Ports in a Virtual Switch . . . . .	30
	Configuring a Layer 2 Virtual Switch with a Layer 2 Trunk Port . . . . .	32
	Configuring a Layer 2 Virtual Switch for MX Series Routers . . . . .	33

<b>Chapter 5</b>	<b>Layer 2 Learning and Forwarding for Bridge Domains . . . . .</b>	<b>37</b>
	Disabling MAC Learning for a Bridge Domain or Logical Interface . . . . .	37
	Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain . . . .	38
	Configuring the Size of the MAC Address Table . . . . .	39
	Limiting MAC Addresses Learned from an Interface in a Bridge Domain . . . . .	40
	Enabling MAC Accounting for a Bridge Domain . . . . .	41
<b>Chapter 6</b>	<b>Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports . . . . .</b>	<b>43</b>
	Disabling MAC Learning for a Set of Bridge Domains . . . . .	43
	Limiting the Number of MAC Addresses Learned from a Trunk Port . . . . .	43
	Configuring the Size of the MAC Address Table for a Set of Bridge Domains . . . .	44
	Enabling MAC Accounting for a Set of Bridge Domains . . . . .	45
<b>Chapter 7</b>	<b>Layer 3 Tunnel Services Interfaces with a DPC . . . . .</b>	<b>47</b>
	Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC . . . . .	47
<b>Chapter 8</b>	<b>Layer 2 Address Learning and Forwarding . . . . .</b>	<b>49</b>
	Configuring the MAC Table Timeout Interval . . . . .	49
	Enabling MAC Accounting . . . . .	50
	Limiting the Number of MAC Addresses Learned from Each Logical Interface . . .	51
	Disabling Layer 2 Learning and Forwarding . . . . .	52
	Example: Loop Detection Using the MAC Move Approach . . . . .	52
<b>Chapter 9</b>	<b>Configuration Statements for Bridge Domains . . . . .</b>	<b>57</b>
	[edit bridge-domains] Hierarchy Level . . . . .	57
	action-priority . . . . .	60
	bridge-domains . . . . .	61
	bridge-options . . . . .	62
	domain-type . . . . .	63
	enable-mac-move-action . . . . .	63
	interface . . . . .	64
	interface-mac-limit . . . . .	65
	mac-statistics . . . . .	67
	mac-table-size . . . . .	69
	mac-table-aging-time . . . . .	71
	no-irb-layer-2-copy . . . . .	72
	no-mac-learning . . . . .	73
	packet-action . . . . .	75
	routing-interface . . . . .	78
	service-id . . . . .	79
	static-mac . . . . .	80
	vlan-id (Bridge Domain or VLAN) . . . . .	81
	vlan-id-list . . . . .	83
	vlan-tags . . . . .	84

<b>Chapter 10</b>	<b>Configuration Statements for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports . . . . .</b>	<b>85</b>
	[edit switch-options] Hierarchy Level . . . . .	85
	switch-options . . . . .	86
	interface-mac-limit . . . . .	87
	mac-statistics . . . . .	89
	mac-table-size . . . . .	91
	no-mac-learning . . . . .	93
	packet-action . . . . .	95
<b>Chapter 11</b>	<b>Configuration Statements for Layer 3 Tunnel Services Interfaces with a DPC . . . . .</b>	<b>99</b>
	tunnel-services (Chassis) . . . . .	100
	bandwidth (Tunnel Services) . . . . .	102
<b>Chapter 12</b>	<b>Configuration Statements for Layer 2 Address Learning and Forwarding . . . . .</b>	<b>105</b>
	[edit protocols l2-learning] Hierarchy Level . . . . .	105
	l2-learning . . . . .	106
	global-mac-limit . . . . .	107
	global-mac-statistics . . . . .	107
	global-mac-table-aging-time . . . . .	108
	global-no-mac-learning . . . . .	108
	interface-mac-limit . . . . .	109
	packet-action . . . . .	111
<b>Part 3</b>	<b>Administration</b>	
<b>Chapter 13</b>	<b>Operational Mode Commands for Bridge Domains . . . . .</b>	<b>117</b>
	clear bridge mac-table . . . . .	118
	clear interfaces mac-database . . . . .	119
	clear interfaces mac-database statistics . . . . .	120
	show bridge domain . . . . .	121
	show bridge flood . . . . .	123
	show bridge mac-table . . . . .	130
	show bridge statistics . . . . .	135
	show interfaces queue . . . . .	138
<b>Chapter 14</b>	<b>Operational Mode Commands for Layer 2 Learning . . . . .</b>	<b>177</b>
	clear l2-learning mac-move-buffer . . . . .	178
	show l2-learning global-information . . . . .	179
	show l2-learning global-mac-count . . . . .	181
	show l2-learning instance . . . . .	182
	show l2-learning interface . . . . .	184
	show l2-learning mac-move-buffer . . . . .	186
<b>Part 4</b>	<b>Index</b>	
	Index . . . . .	189



# List of Figures

Part 2	Configuration	
Chapter 3	Layer 2 Bridging and Layer 3 IP Routing . . . . .	11
	Figure 1: Basic Layer 2 Switching . . . . .	13





# List of Tables

	<b>About the Documentation</b> . . . . .	<b>xi</b>
	Table 1: Notice Icons . . . . .	xiii
	Table 2: Text and Syntax Conventions . . . . .	xiii
<b>Part 2</b>	<b>Configuration</b>	
<b>Chapter 3</b>	<b>Layer 2 Bridging and Layer 3 IP Routing</b> . . . . .	<b>11</b>
	Table 3: Statement Usage and Input Rewrite Operations for VLAN Identifiers for a Bridge Domain . . . . .	23
	Table 4: Statement Usage and Output Rewrite Operations for VLAN Identifiers for a Bridge Domain . . . . .	24
<b>Part 3</b>	<b>Administration</b>	
<b>Chapter 13</b>	<b>Operational Mode Commands for Bridge Domains</b> . . . . .	<b>117</b>
	Table 5: show bridge mac-table Output fields . . . . .	131
	Table 6: Layer 2 Overhead and Transmitted Packets or Byte Counts . . . . .	139
	Table 7: show interfaces queue Output Fields . . . . .	142
	Table 8: Byte Count by Interface Hardware . . . . .	145
<b>Chapter 14</b>	<b>Operational Mode Commands for Layer 2 Learning</b> . . . . .	<b>177</b>
	Table 9: show l2-learning global-information Output Fields . . . . .	179
	Table 10: show l2-learning instance Output Fields . . . . .	182
	Table 11: show l2-learning interfaceOutput Fields . . . . .	184



# About the Documentation

- Documentation and Release Notes on page xi
- Supported Platforms on page xi
- Using the Examples in This Manual on page xi
- Documentation Conventions on page xiii
- Documentation Feedback on page xv
- Requesting Technical Support on page xv

## Documentation and Release Notes

---

To obtain the most current version of all Juniper Networks<sup>®</sup> technical documentation, see the product documentation page on the Juniper Networks website at <http://www.juniper.net/techpubs/>.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at <http://www.juniper.net/books>.

## Supported Platforms

---

For the features described in this document, the following platforms are supported:

- MX Series

## Using the Examples in This Manual

---

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

## Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

## Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

```
commit {
  file ex-script-snippet.xml; }
```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the *CLI User Guide*.

## Documentation Conventions

Table 1 on page xiii defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page xiii defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
<b>Bold text like this</b>	Represents text that you type.	To enter configuration mode, type the <b>configure</b> command:  user@host> <b>configure</b>

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> <b>show chassis alarms</b>  No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"><li>Introduces or emphasizes important new terms.</li><li>Identifies guide names.</li><li>Identifies RFC and Internet draft titles.</li></ul>	<ul style="list-style-type: none"><li>A policy <i>term</i> is a named structure that defines match conditions and actions.</li><li><i>Junos OS CLI User Guide</i></li><li>RFC 1997, <i>BGP Communities Attribute</i></li></ul>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name:  [edit] root@# <b>set system domain-name</b> <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"><li>To configure a stub area, include the <b>stub</b> statement at the [edit protocols <b>ospf area area-id</b>] hierarchy level.</li><li>The console port is labeled <b>CONSOLE</b>.</li></ul>
< > (angle brackets)	Encloses optional keywords or variables.	<b>stub &lt;default-metric <i>metric</i>&gt;;</b>
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	<b>broadcast   multicast</b>  <b>(<i>string1</i>   <i>string2</i>   <i>string3</i>)</b>
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	<b>rsvp { # Required for dynamic MPLS only</b>
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	<b>community name members [</b> <i>community-ids</i> <b>]</b>
Indentation and braces ( { } )	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none"><li>In the Logical Interfaces box, select <b>All Interfaces</b>.</li><li>To cancel the configuration, click <b>Cancel</b>.</li></ul>

Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select <b>Protocols&gt;Ospf</b> .

## Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page at the Juniper Networks Technical Documentation site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>.
- E-mail—Send your comments to [techpubs-comments@juniper.net](mailto:techpubs-comments@juniper.net). Include the document or topic name, URL or page number, and software version (if applicable).

## Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

## Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>

- Download the latest versions of software and review release notes:  
<http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications:  
<http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum:  
<http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://tools.juniper.net/SerialNumberEntitlementSearch/>

## Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html>.



## PART 1

# Overview

- [Layer 2 Bridging on page 3](#)
- [Layer 2 Address Learning and Forwarding on page 7](#)



## CHAPTER 1

# Layer 2 Bridging

- [Layer 2 Bridge Domains Overview on page 3](#)
- [Layer 2 Virtual Switches Overview on page 4](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- [Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 6](#)

### Layer 2 Bridge Domains Overview

---

You can configure one or more bridge domains on MX Series routers to perform Layer 2 bridging. The Layer 2 bridging functions of the MX Series routers include integrated routing and bridging (IRB) for support for Layer 2 bridging and Layer 3 IP routing on the same interface, and virtual switches that isolate a LAN segment with its spanning-tree protocol instance and separate its VLAN ID space.

A bridge domain is a set of logical ports that share the same flooding or broadcast characteristics. Like a virtual LAN (VLAN), a bridge domain spans one or more ports of multiple devices.

On Juniper Networks MX Series 3D Universal Edge Routers only, you can configure one or more bridge domains to perform Layer 2 bridging. Thus, MX Series routers can function as Layer 2 switches, each with multiple bridging, or broadcast, domains that participate in the same Layer 2 network. You can also configure Layer 3 routing support for a bridge domain. Integrated routing and bridging (IRB) provides support for Layer 2 bridging and Layer 3 IP routing on the same interface. IRB enables you to route packets to another routed interface or to another bridge domain that has a Layer 3 protocol configured.

You can also group one or more bridge domains within a single instance, or virtual switch. The MX Series routers also support multiple virtual switches, each of which operates independently of other virtual switches on the router. Virtual switches isolate a LAN segment with its spanning-tree protocol instance and separate its VLAN ID space. Thus, each virtual switch can participate in a different Layer 2 network.

In Junos OS Release 9.2 and later, bridge domains provide support for a Layer 2 trunk port. A Layer 2 trunk interface enables you to configure a single logical interface to represent multiple VLANs on a physical interface. You can configure a set of bridge

domains and VLAN identifiers that are automatically associated with one or more Layer 2 trunk interfaces. Packets received on a trunk interface are forwarded within a bridge domain that has the same VLAN identifier. A Layer 2 trunk interface also supports IRB within a bridge domain. In addition, you can configure Layer 2 learning and forwarding properties that apply to the entire set of bridge domains.

In Junos OS Release 9.3 and later, you can configure VPLS ports in a virtual switch instead of a dedicated routing instance of type **vpls** so that the logical interfaces of the Layer 2 bridge domains in the virtual switch can handle VPLS routing instance traffic. Packets received on a Layer 2 trunk interface are forwarded within a bridge domain that has the same VLAN identifier.

**Related  
Documentation**

- [Layer 2 Virtual Switches Overview on page 4](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- [Configuring a Bridge Domain on page 11](#)

---

## Layer 2 Virtual Switches Overview

---

On MX Series routers only, you can group one or more bridge domains to form a virtual switch to isolate a LAN segment with its spanning-tree protocol instance and separate its VLAN ID space. A bridge domain consists of a set of logical ports that share the same flooding or broadcast characteristics. Like a virtual LAN, a bridge domain spans one or more ports of multiple devices. You can configure multiple virtual switches, each of which operates independently of the other virtual switches on the routing platform. Thus, each virtual switch can participate in a different Layer 2 network.

You can configure a virtual switch to participate only in Layer 2 bridging and optionally to perform Layer 3 routing. In addition, you can configure one of three Layer 2 control protocols—Spanning-Tree Protocol, Rapid Spanning-Tree Protocol (RSTP), or Multiple Spanning-Tree Protocol (MSTP)—to prevent forwarding loops. For more information about how to configure Layer 2 logical ports on an interface, see the *Junos OS Network Interfaces Library for Routing Devices*.

In Junos OS Release 9.2 and later, you can associate one or more logical interfaces configured as trunk interfaces with a virtual switch. A trunk interface, or Layer 2 trunk port, enables you to configure a logical interface to represent multiple VLANs on the physical interface. Packets received on a trunk interface are forwarded within a bridge domain that has same VLAN identifier. For more information about how to configure trunk interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

You can also configure Layer 2 forwarding and learning properties for the virtual switch as well as any bridge domains that belong to a virtual switch. .

For more information about configuring a routing instance for Layer 2 VPN, see the *Junos OS VPNs Library for Routing Devices*. For a detailed Layer 2 VPN example configuration, see the *Junos OS, Release 15.1*.

**Related  
Documentation**

- [Configuring a Layer 2 Control Protocol Routing Instance](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- [Layer 2 Protocol Tunneling Through a Network Overview](#)

## Layer 2 Learning and Forwarding for Bridge Domains Overview

When you configure a bridge domain, Layer 2 address learning is enabled by default. The bridge domain learns unicast media access control (MAC) addresses to avoid flooding the packets to all the ports in the bridge domain. Each bridge domain creates a source MAC entry in its source and destination MAC tables for each source MAC address learned from packets received on the ports that belong to the bridge domain.



**NOTE:** Traffic is not flooded back onto the interface on which it was received. However, because this “split horizon” occurs at a late stage, the packet statistics displayed by commands such as `show interfaces queue` will include flood traffic.

You can optionally disable MAC learning either for the entire router or for a specific bridge domain or logical interface. You can also configure the following Layer 2 learning and forwarding properties:

- Static MAC entries for logical interfaces only
- Limit to the number of MAC addresses learned from a specific logical interface or from all the logical interfaces in a bridge domain
- Size of the MAC address table for the bridge domain
- MAC accounting for a bridge domain

**Related  
Documentation**

- [Layer 2 Learning and Forwarding Overview on page 7](#)

## Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports

Layer 2 learning is enabled by default. A set of bridge domains, configured to function as a switch with a Layer 2 trunk port, learns unicast media access control (MAC) addresses to avoid flooding packets to the trunk port.



**NOTE:** Traffic is not flooded back onto the interface on which it was received. However, because this “split horizon” occurs at a late stage, the packet statistics displayed by commands such as `show interfaces queue` will include flood traffic.

You can optionally disable Layer 2 learning for the entire set of bridge domains as well as modify the following Layer 2 learning and forwarding properties:

- Limit the number of MAC addresses learned from the Layer 2 trunk port associated with the set of bridge domains
- Modify the size of the MAC address table for the set of bridge domains
- Enable MAC accounting for the set of bridge domains

**Related  
Documentation**

- [Layer 2 Learning and Forwarding Overview on page 7](#)

## Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances

---

For a bridge domain that is performing Layer 2 switching only, you do not have to specify a VLAN identifier.

For a bridge domain that is performing Layer 3 IP routing, you must specify either a VLAN identifier or dual VLAN identifier tags.

For a VPLS routing instance, you must specify either a VLAN identifier or dual VLAN identifier tags.

**Related  
Documentation**

- [Layer 2 Learning and Forwarding Overview on page 7](#)

## CHAPTER 2

# Layer 2 Address Learning and Forwarding

- [Layer 2 Learning and Forwarding Overview on page 7](#)

## Layer 2 Learning and Forwarding Overview

---

On MX Series routers only, you can configure Layer 2 MAC address and VLAN learning and forwarding properties in support of Layer 2 bridging. The router learns unicast media access control (MAC) addresses to avoid flooding the packets to all the ports in a bridge domain. The MX Series router creates a source MAC entry in its source and destination MAC tables for each MAC address learned from packets received on ports that belong to the bridge domain. If the bridge domain receives a control protocol data unit (PDU) which does not have a corresponding protocol configured, then the control PDU is considered as an unknown multicast data packet and the packets are flooded across all the ports that are part of the same bridge domain. If the bridge domain has the protocol corresponding to the PDU configured, then the control PDU is considered as a control packet and is processed by the routing engine.

By default, Layer 2 address learning is enabled. You can disable MAC learning for the router or for a specific bridge domain or logical interfaces. You can also configure the following Layer 2 forwarding properties for an MX Series router:

- Timeout interval for MAC entries
- MAC accounting
- A limit to the number of MAC addresses learned from the logical interfaces

For more information about how to configure bridge domains and virtual switches, see [“Configuring a Bridge Domain” on page 11](#) and [“Configuring a Layer 2 Virtual Switch for MX Series Routers” on page 33](#).

### Related Documentation

- [Configuring the MAC Table Timeout Interval on page 49](#)
- [Enabling MAC Accounting on page 50](#)
- [Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 51](#)
- [Disabling Layer 2 Learning and Forwarding on page 52](#)





## PART 2

# Configuration

- [Layer 2 Bridging and Layer 3 IP Routing on page 11](#)
- [Layer 2 Virtual Switches on page 29](#)
- [Layer 2 Learning and Forwarding for Bridge Domains on page 37](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 43](#)
- [Layer 3 Tunnel Services Interfaces with a DPC on page 47](#)
- [Layer 2 Address Learning and Forwarding on page 49](#)
- [Configuration Statements for Bridge Domains on page 57](#)
- [Configuration Statements for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 85](#)
- [Configuration Statements for Layer 3 Tunnel Services Interfaces with a DPC on page 99](#)
- [Configuration Statements for Layer 2 Address Learning and Forwarding on page 105](#)



## CHAPTER 3

# Layer 2 Bridging and Layer 3 IP Routing

- [Configuring a Bridge Domain on page 11](#)
- [Example: Configuring Basic Layer 2 Switching on MX Series on page 12](#)
- [Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19](#)
- [Configuring Integrated Routing and Bridging for Bridge Domains on page 24](#)
- [Configuring Bridge Domains as Switches for Layer 2 Trunk Ports on page 26](#)

## Configuring a Bridge Domain

---

A bridge domain must include a set of logical interfaces that participate in Layer 2 learning and forwarding. You can optionally configure a VLAN identifier and a routing interface for the bridge domain to also support Layer 3 IP routing.

To enable a bridge domain, include the following statements:

```
[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    interface interface-name;
    routing-interface routing-interface-name;
    vlan-id (none | all | number);
    vlan-id-list [ vlan-id-numbers ];
    vlan-tags outer number inner number);
  }
}
```

You cannot use the slash (/) character in bridge domain names. If you do, the configuration does not commit and an error is generated.

For the **vlan-id** statement, you can specify either a valid VLAN identifier or the **none** or **all** options. For information about VLAN identifiers and VLAN tags for a bridge domain, see [“Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances” on page 6](#) and [“Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances” on page 19](#).

To include one or more logical interfaces in the bridge domain, specify an **interface-name** for an Ethernet interface you configured at the **[edit interfaces]** hierarchy level.



**NOTE:** A maximum of 4000 active logical interfaces are supported on a bridge domain or on each mesh group in a virtual private LAN service (VPLS) instance configured for Layer 2 bridging.

By default, each bridge domain maintains a Layer 2 forwarding database that contains media access control (MAC) addresses learned from packets received on the ports that belong to the bridge domain. You can modify Layer 2 forwarding properties, including disabling MAC learning for the entire system or a bridge domain, adding static MAC addresses for specific logical interfaces, and limiting the number of MAC addresses learned by the entire system, the bridge domain, or a logical interface.

You can also configure spanning tree protocols to prevent forwarding loops. For more information, see *Spanning-Tree Protocols Supported*.

In Junos OS Release 8.5 and later, you can configure IGMP snooping for a bridge domain. For more information, see the *Multicast Protocols Feature Guide for Routing Devices*.

**Related  
Documentation**

- [Configuring Integrated Routing and Bridging for Bridge Domains on page 24](#)
- [Layer 2 Learning and Forwarding Overview on page 7](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)

---

## Example: Configuring Basic Layer 2 Switching on MX Series

This example shows how to configure Layer 2 switching with all interfaces participating in a single VLAN.

- [Requirements on page 12](#)
- [Overview on page 12](#)
- [Configuration on page 13](#)
- [Verification on page 15](#)

### Requirements

No special configuration beyond device initialization is required before configuring this example.

This example uses an MX Series device to perform Layer 2 switching.

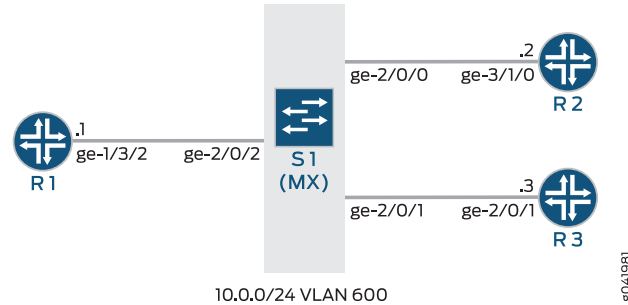
### Overview

In this example, a single MX Series device is configured to act as a basic single-VLAN switch. Three connections are in place. The connections from the MX Series device attach to Junos OS routers, but the routers are used here for testing purposes only. In place of routers, you can use any IP networking devices.

## Topology

Figure 1 on page 13 shows the sample network.

Figure 1: Basic Layer 2 Switching



“CLI Quick Configuration” on page 13 shows the configuration for all of the devices in Figure 1 on page 13.

The section “Step-by-Step Procedure” on page 14 describes the steps on Device S1.

## Configuration

<b>CLI Quick Configuration</b>	To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the <b>[edit]</b> hierarchy level.
<b>Device S1</b>	<pre> set interfaces ge-2/0/0 vlan-tagging set interfaces ge-2/0/0 encapsulation extended-vlan-bridge set interfaces ge-2/0/0 unit 0 vlan-id 600 set interfaces ge-2/0/1 vlan-tagging set interfaces ge-2/0/1 encapsulation extended-vlan-bridge set interfaces ge-2/0/1 unit 0 vlan-id 600 set interfaces ge-2/0/2 vlan-tagging set interfaces ge-2/0/2 encapsulation extended-vlan-bridge set interfaces ge-2/0/2 unit 0 vlan-id 600 set bridge-domains customer1 domain-type bridge set bridge-domains customer1 interface ge-2/0/0.0 set bridge-domains customer1 interface ge-2/0/2.0 set bridge-domains customer1 interface ge-2/0/1.0 </pre>
<b>Device R1</b>	<pre> set interfaces ge-1/3/2 vlan-tagging set interfaces ge-1/3/2 unit 0 vlan-id 600 set interfaces ge-1/3/2 unit 0 family inet address 10.0.0.1/24 </pre>
<b>Device R2</b>	<pre> set interfaces ge-3/1/0 vlan-tagging set interfaces ge-3/1/0 unit 0 vlan-id 600 set interfaces ge-3/1/0 unit 0 family inet address 10.0.0.2/24 </pre>
<b>Device R3</b>	<pre> set interfaces ge-2/0/1 vlan-tagging set interfaces ge-2/0/1 unit 0 vlan-id 600 set interfaces ge-2/0/1 unit 0 family inet address 10.0.0.3/24 </pre>

**Step-by-Step Procedure** The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device S1:

1. Configure the device interfaces.

```
[edit interfaces]
user@S1# set interfaces ge-2/0/0 vlan-tagging
user@S1# set interfaces ge-2/0/0 encapsulation extended-vlan-bridge
user@S1# set interfaces ge-2/0/0 unit 0 vlan-id 600
```

```
user@S1# set interfaces ge-2/0/1 vlan-tagging
user@S1# set interfaces ge-2/0/1 encapsulation extended-vlan-bridge
user@S1# set interfaces ge-2/0/1 unit 0 vlan-id 600
```

```
user@S1# set interfaces ge-2/0/2 vlan-tagging
user@S1# set interfaces ge-2/0/2 encapsulation extended-vlan-bridge
user@S1# set interfaces ge-2/0/2 unit 0 vlan-id 600
```

2. Configure the bridge domain.

```
[edit interfaces]
user@S1# set bridge-domains customer1 domain-type bridge
user@S1# set bridge-domains customer1 interface ge-2/0/0.0
user@S1# set bridge-domains customer1 interface ge-2/0/2.0
user@S1# set bridge-domains customer1 interface ge-2/0/1.0
```

**Results** From configuration mode, confirm your configuration by entering the **show interfaces** and **show bridge-domains** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@S1# show interfaces
ge-2/0/0 {
  vlan-tagging;
  encapsulation extended-vlan-bridge;
  unit 0 {
    vlan-id 600;
  }
}
ge-2/0/1 {
  vlan-tagging;
  encapsulation extended-vlan-bridge;
  unit 0 {
    vlan-id 600;
  }
}
ge-2/0/2 {
  vlan-tagging;
  encapsulation extended-vlan-bridge;
  unit 0 {
    vlan-id 600;
  }
}
```

```

user@S1# show bridge-domains
customer1 {
  domain-type bridge;
  interface ge-2/0/0.0;
  interface ge-2/0/2.0;
  interface ge-2/0/1.0;
}

```

If you are done configuring the device, enter **commit** from configuration mode.

## Verification

Confirm that the configuration is working properly.

- [Confirming the MAC Address Learning on page 15](#)
- [Making Sure That the Attached Devices Can Reach Each Other on page 16](#)
- [Checking the Bridge Domain on page 16](#)
- [Checking the Bridge Statistics on page 17](#)
- [Checking the Bridge Flooding on page 17](#)
- [Checking Layer 2 Learning on page 18](#)

### Confirming the MAC Address Learning

**Purpose** Display Layer 2 MAC address information.

**Action** • From Device S1, run the **show bridge mac-table** command.

```
user@S1> show bridge mac-table
```

MAC flags (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC  
SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)

```

Routing instance : default-switch
Bridging domain : customer1, VLAN : NA
  MAC          MAC      Logical      NH      RTR
  address      flags    interface  Index   ID
  00:12:1e:ee:34:dd  D      ge-2/0/2.0
  00:1d:b5:5e:86:79  D      ge-2/0/0.0
  00:21:59:0f:35:2b  D      ge-2/0/1.0

```

- From Device S1, run the **show bridge mac-table extensive** command.

```
user@S1> show bridge mac-table extensive
```

```

MAC address: 00:12:1e:ee:34:dd
Routing instance: default-switch
Bridging domain: customer1, VLAN : NA
Learning interface: ge-2/0/2.0
Layer 2 flags: in_hash,in_ifd,in_ifl,in_vlan,in_rtt,kernel,in_ifbd
Epoch: 1                               Sequence number: 0
Learning mask: 0x00000004

```

```

MAC address: 00:1d:b5:5e:86:79
Routing instance: default-switch
Bridging domain: customer1, VLAN : NA
Learning interface: ge-2/0/0.0

```

```
Layer 2 flags: in_hash,in_ifd,in_ifl,in_vlan,in_rtt,kernel,in_ifbd
Epoch: 1                               Sequence number: 0
Learning mask: 0x00000004
```

```
MAC address: 00:21:59:0f:35:2b
Routing instance: default-switch
Bridging domain: customer1, VLAN : NA
Learning interface: ge-2/0/1.0
Layer 2 flags: in_hash,in_ifd,in_ifl,in_vlan,in_rtt,kernel,in_ifbd
Epoch: 3                               Sequence number: 0
Learning mask: 0x00000004
```

**Meaning** The output shows that the MAC addresses have been learned.

---

### Making Sure That the Attached Devices Can Reach Each Other

**Purpose** Verify connectivity.

**Action**

```
user@R1> ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2): 56 data bytes
64 bytes from 10.0.0.2: icmp_seq=0 ttl=64 time=1.178 ms
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=1.192 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=1.149 ms
^C
--- 10.0.0.2 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.149/1.173/1.192/0.018 ms
```

```
user@R1> ping 10.0.0.3
PING 10.0.0.3 (10.0.0.3): 56 data bytes
64 bytes from 10.0.0.3: icmp_seq=0 ttl=64 time=1.189 ms
64 bytes from 10.0.0.3: icmp_seq=1 ttl=64 time=1.175 ms
64 bytes from 10.0.0.3: icmp_seq=2 ttl=64 time=1.178 ms
64 bytes from 10.0.0.3: icmp_seq=3 ttl=64 time=1.133 ms
^C
--- 10.0.0.3 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 1.133/1.169/1.189/0.021 ms
```

```
user@R2> ping 10.0.0.3
PING 10.0.0.3 (10.0.0.3): 56 data bytes
64 bytes from 10.0.0.3: icmp_seq=0 ttl=64 time=0.762 ms
64 bytes from 10.0.0.3: icmp_seq=1 ttl=64 time=0.651 ms
64 bytes from 10.0.0.3: icmp_seq=2 ttl=64 time=0.722 ms
64 bytes from 10.0.0.3: icmp_seq=3 ttl=64 time=0.705 ms
^C
--- 10.0.0.3 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.651/0.710/0.762/0.040 ms
```

**Meaning** The output shows that the attached devices have established Layer 3 connectivity, with Device S1 doing transparent Layer 2 bridging.

---

### Checking the Bridge Domain

**Purpose** Display bridge domain information.



**Action** user@S1> `show bridge domain extensive`

```
Routing instance: default-switch
Bridge domain: customer1                      State: Active
Bridge VLAN ID: NA
Interfaces:
  ge-2/0/0.0
  ge-2/0/1.0
  ge-2/0/2.0
Total MAC count: 3
```

**Meaning** The output shows that bridge domain is active.

### Checking the Bridge Statistics

**Purpose** Display bridge statistics.

**Action** user@S1> `show bridge statistics`

```
Local interface: ge-2/0/0.0, Index: 65543
Broadcast packets:          0
Broadcast bytes :           0
Multicast packets:         80
Multicast bytes :         8160
Flooded packets :           0
Flooded bytes :            0
Unicast packets :           1
Unicast bytes :            64
Current MAC count:         1 (Limit 1024)
Local interface: ge-2/0/2.0, Index: 324
Broadcast packets:          0
Broadcast bytes :           0
Multicast packets:         80
Multicast bytes :         8160
Flooded packets :           1
Flooded bytes :            74
Unicast packets :           52
Unicast bytes :          4332
Current MAC count:         1 (Limit 1024)
Local interface: ge-2/0/1.0, Index: 196613
Broadcast packets:          2
Broadcast bytes :          128
Multicast packets:          0
Multicast bytes :           0
Flooded packets :           1
Flooded bytes :            93
Unicast packets :           51
Unicast bytes :          4249
Current MAC count:         1 (Limit 1024)
```

**Meaning** The output shows that bridge domain interfaces are sending and receiving packets.

### Checking the Bridge Flooding

**Purpose** Display bridge flooding information.

**Action** user@S1> `show bridge flood extensive`

```
Name: __juniper_private1__
CEs: 0
VEs: 0
Name: default-switch
CEs: 3
VEs: 0
Bridging domain: customer1
  Flood route prefix: 0x30003/51
  Flood route type: FLOOD_GRP_COMP_NH
  Flood route owner: __all_ces__
  Flood group name: __all_ces__
  Flood group index: 1
  Nexthop type: comp
  Nexthop index: 568
  Flooding to:
    Name          Type          NhType      Index
    __all_ces__   Group          comp        562
    Composition: split-horizon
    Flooding to:
      Name          Type          NhType      Index
      ge-2/0/0.0    CE            ucst        524
      ge-2/0/1.0    CE            ucst        513
      ge-2/0/2.0    CE            ucst        523

  Flood route prefix: 0x30005/51
  Flood route type: FLOOD_GRP_COMP_NH
  Flood route owner: __re_flood__
  Flood group name: __re_flood__
  Flood group index: 65534
  Nexthop type: comp
  Nexthop index: 565
  Flooding to:
    Name          Type          NhType      Index
    __all_ces__   Group          comp        562
    Composition: split-horizon
    Flooding to:
      Name          Type          NhType      Index
      ge-2/0/0.0    CE            ucst        524
      ge-2/0/1.0    CE            ucst        513
      ge-2/0/2.0    CE            ucst        523
```

**Meaning** If the destination MAC address of a packet is unknown to the device (that is, the destination MAC address in the packet does not have an entry in the forwarding table), the device duplicates the packet and floods it on all interfaces in the bridge domain other than the interface on which the packet arrived. This is known as packet flooding and is the default behavior for the device to determine the outgoing interface for an unknown destination MAC address.

### Checking Layer 2 Learning

**Purpose** Display Layer 2 learning information for all the interfaces.

**Action** user@S1> `show l2-learning interface`

```

Routing Instance Name : default-switch
Logical Interface flags (DL -disable learning, AD -packet action drop,
                        LH - MAC limit hit, DN - Interface Down )
Logical      BD      MAC      STP      Logical
Interface    Name    Limit    State   Interface flags
ge-2/0/2.0   custom.. 1024    Forwarding

Routing Instance Name : default-switch
Logical Interface flags (DL -disable learning, AD -packet action drop,
                        LH - MAC limit hit, DN - Interface Down )
Logical      BD      MAC      STP      Logical
Interface    Name    Limit    State   Interface flags
ge-2/0/0.0   custom.. 1024    Forwarding

Routing Instance Name : default-switch
Logical Interface flags (DL -disable learning, AD -packet action drop,
                        LH - MAC limit hit, DN - Interface Down )
Logical      BD      MAC      STP      Logical
Interface    Name    Limit    State   Interface flags
ge-2/0/1.0   custom.. 1024    Forwarding

```

**Related Documentation**

- *Understanding OSPF Areas*
- *Examples: Configuring OSPF Stub and Not-So-Stubby Areas*

## Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances

You can configure VLAN identifiers for a bridge domain or a VPLS routing instance in the following ways:

- By using the **input-vlan-map** and the **output-vlan-map** statements at the **[edit interfaces interface-name]** or **[edit logical-systems logical-system-name interfaces interface-name]** hierarchy level to configure VLAN mapping. For information about configuring input and output VLAN maps to stack and rewrite VLAN tags in incoming or outgoing frames, see the *Junos OS Network Interfaces Library for Routing Devices*.
- By using either the **vlan-id** statement or the **vlan-tags** statement to configure a normalizing VLAN identifier. This topic describes how normalizing VLAN identifiers are processed and translated in a bridge domain or a VPLS routing instance.

The **vlan-id** and **vlan-tags** statements are used to specify the normalizing VLAN identifier under the bridge domain or VPLS routing instance. The normalizing VLAN identifier is used to perform the following functions:

- Translate, or normalize, the VLAN tags of received packets received into a learn VLAN identifier.
- Create multiple learning domains that each contain a learn VLAN identifier. A learning domain is a MAC address database to which MAC addresses are added based on the learn VLAN identifier.



**NOTE:** You cannot configure VLAN mapping using the `input-vlan-map` and `output-vlan-map` statements if you configure a normalizing VLAN identifier for a bridge domain or VPLS routing instance using the `vlan-id` or `vlan-tags` statements.

To configure a VLAN identifier for a bridge domain, include either the `vlan-id` or the `vlan-tags` statement at the `[edit interfaces interface-name unit logic-unit-number family bridge]` or `[edit logical-systems logical-system-name interfaces interface-name unit logic-unit-number family bridge]` hierarchy level, and then include that logical interface in the bridge domain configuration. For more information about configuring a bridge domain, see [“Configuring a Bridge Domain” on page 11](#).

For a VPLS routing instance, include either the `vlan-id` or `vlan-tags` statement at the `[edit interfaces interface-name unit logic-unit-number]` or `[edit logical-systems logical-system-name interfaces interface-name unit logic-unit-number]` hierarchy level, and then include that logical interface in the VPLS routing instance configuration. For more information about configuring a VPLS routing instance, see the *Junos OS VPNs Library for Routing Devices*.



**NOTE:** The maximum number of Layer 2 interfaces that you can associate with a bridge domain or a VPLS instance on MX Series routers is 4000.



**NOTE:** For a single bridge domain or VPLS routing instance, you can include either the `vlan-id` or the `vlan-tags` statement, but not both. If you do not configure a `vlan-id`, `vlan-tags`, or `vlan-id-list [ vlan-id-numbers ]` for the bridge domain or the VPLS routing instance, the Layer 2 packets received are forwarded to the outbound Layer 2 interface without having the VLAN tag modified unless an `output-vlan-map` is configured on the Layer 2 interface. This results in a frame being forwarded to a Layer 2 interface with a VLAN tag that is different from what is configured for the Layer 2 interface. Note that a frame received from the Layer 2 interface is still required to match the VLAN tag(s) specified in the interface configuration. The invalid configuration may cause a Layer 2 loop to occur.

The VLAN tags associated with the inbound logical interface are compared with the normalizing VLAN identifier. If the tags are different, they are rewritten as described in [Table 3 on page 23](#). The source MAC address of a received packet is learned based on the normalizing VLAN identifier.



**NOTE:** You do not have to specify a VLAN identifier for a bridge domain that is performing Layer 2 switching only. To support Layer 3 IP routing, you must specify either a VLAN identifier or a pair of VLAN tags. However, you cannot specify the same VLAN identifier for more than one bridge domain within a routing instance. Each bridge domain must have a unique VLAN identifier.

If the VLAN tags associated with the outbound logical interface and the normalizing VLAN identifier are different, the normalizing VLAN identifier is rewritten to match the VLAN tags of the outbound logical interface, as described in [Table 4 on page 24](#).

For the packets sent over the VPLS routing instance to be tagged by the normalizing VLAN identifier, include one of the following configuration statements:

- **vlan-id *number*** to tag all packets that are sent over the VPLS virtual tunnel (VT) interfaces with the VLAN identifier.
- **vlan-tags outer *number* inner *number*** to tag all packets sent over the VPLS VT interfaces with dual outer and inner VLAN tags.

Use the **vlan-id none** statement to have the VLAN tags removed from packets associated with an inbound logical interface when those packets are sent over VPLS VT interfaces. Note that those packets might still be sent with other customer VLAN tags.

The **vlan-id all** statement enables you to configure bridging for several VLANs with a minimum amount of configuration. Configuring this statement creates a learning domain for:

- Each inner VLAN, or learn VLAN, identifier of a logical interface configured with two VLAN tags
- Each VLAN, or learn VLAN, identifier of a logical interface configured with one VLAN tag

We recommend that you do not use customer VLAN IDs in a VPLS routing instance because customer VLAN IDs are used for learning only.

You should use the service VLAN ID in a VPLS routing instance, as in the following configuration:

```
[edit]
interface ge-1/1/1 {
  vlan-tagging;
  unit 1 {
    vlan-id s1; /* Service vlan */
    encapsulation vlan-vpls;
    input-vlan-map pop; /* Pop the service vlan on input */
    output-vlan-map push; /* Push the service vlan on output */
  }
}
interface ge-1/1/2 {
  encapsulation ethernet-vpls;
  unit 0;
```

```

}
routing-instance {
  V1 {
    instance-type vpls;
    vlan-id all;
    interface ge-1/1/1.1;
    interface ge-1/1/2.0;
  }
}

```



**NOTE:** If you configure the `vlan-id all` statement in a VPLS routing instance, we recommend using the `input-vlan-map pop` and `output-vlan-map push` statements on the logical interface to pop the service VLAN ID on input and push the service VLAN ID on output and in this way limit the impact of doubly-tagged frames on scaling. You cannot use the native `vlan-id` statement when the `vlan-id all` statement is included in the configuration.

The `vlan-id-list [ vlan-id-numbers ]` statement enables you to configure bridging for multiple VLANs on a trunk interface. Configuring this statement creates a learning domain for:

- Each VLAN listed: `vlan-id-list [ 100 200 300 ]`
- Each VLAN in a range: `vlan-id-list [ 100-200 ]`
- Each VLAN in a list and range combination: `vlan-id-list [ 50, 100-200, 300 ]`

The following steps outline the process for bridging a packet received over a Layer 2 logical interface when you specify a normalizing VLAN identifier using either the `vlan-id number` or `vlan-tags` statement for a bridge domain or a VPLS routing instance:

1. When a packet is received on a physical port, it is accepted only if the VLAN identifier of the packet matches the VLAN identifier of one of the logical interfaces configured on that port.
2. The VLAN tags of the received packet are then compared with the normalizing VLAN identifier. If the VLAN tags of the packet are different from the normalizing VLAN identifier, the VLAN tags are rewritten as described in [Table 3 on page 23](#).
3. If the source MAC address of the received packet is not present in the source MAC table, it is learned based on the normalizing VLAN identifier.
4. The packet is then forwarded toward one or more outbound Layer 2 logical interfaces based on the destination MAC address. A packet with a known unicast destination MAC address is forwarded only to one outbound logical interface. For each outbound Layer 2 logical interface, the normalizing VLAN identifier configured for the bridge domain or VPLS routing instance is compared with the VLAN tags configured on that logical interface. If the VLAN tags associated with an outbound logical interface do not match the normalizing VLAN identifier configured for the bridge domain or VPLS routing instance, the VLAN tags are rewritten as described in [Table 4 on page 24](#).

The tables below show how VLAN tags are applied for traffic sent to and from the bridge domain, depending on how the `vlan-id` and `vlan-tags` statements are configured for the

bridge domain and on how VLAN identifiers are configured for the logical interfaces in a bridge domain or VPLS routing instance. Depending on your configuration, the following rewrite operations are performed on VLAN tags:

- **pop**—Remove a VLAN tag from the top of the VLAN tag stack.
- **pop-pop**—Remove both the outer and inner VLAN tags of the frame.
- **pop-swap**—Remove the outer VLAN tag of the frame and replace the inner VLAN tag of the frame.
- **swap**—Replace the VLAN tag of the frame.
- **push**—Add a new VLAN tag to the top of the VLAN stack.
- **push-push**—Push two VLAN tags in front of the frame.
- **swap-push**—Replace the VLAN tag of the frame and add a new VLAN tag to the top of the VLAN stack.
- **swap-swap**—Replace both the outer and inner VLAN tags of the frame.

Table 3 on page 23 shows specific examples of how the VLAN tags for packets sent to the bridge domain are processed and translated, depending on your configuration. “–” means that the statement is not supported for the specified logical interface VLAN identifier. “No operation” means that the VLAN tags of the received packet are not translated for the specified input logical interface.

**Table 3: Statement Usage and Input Rewrite Operations for VLAN Identifiers for a Bridge Domain**

VLAN Identifier of Logical Interface	VLAN Configurations for Bridge Domain			
	vlan-id none	vlan-id 200	vlan-id all	vlan tags outer 100 inner 300
none	No operation	push 200	–	push 100, push 300
200	pop 200	No operation	No operation	swap 200 to 300, push 100
1000	pop 1000	swap 1000 to 200	No operation	swap 1000 to 300, push 100
vlan-tags outer 2000 inner 300	pop 2000, pop 300	pop 2000, swap 300 to 200	pop 2000	swap 2000 to 100
vlan-tags outer 100 inner 400	pop 100, pop 400	pop 100, swap 400 to 200	pop 100	swap 400 to 300
vlan-id-range 10-100	–	–	No operation	–
vlan-tags outer 200 inner-range 10-100	–	–	pop 200	–

Table 4 on page 24 shows specific examples of how the VLAN tags for packets sent from the bridge domain are processed and translated, depending on your configuration. “–” means that the statement is not supported for the specified logical interface VLAN identifier. “No operation” means that the VLAN tags of the outbound packet are not translated for the specified output logical interface.

**Table 4: Statement Usage and Output Rewrite Operations for VLAN Identifiers for a Bridge Domain**

VLAN Identifier of Logical Interface	VLAN Configurations for Bridge Domain			
	vlan-id none	vlan-id 200	vlan-id all	vlan tags outer 100 inner 300
none	no operation	pop 200	–	pop 100, pop 300
200	push 200	No operation	No operation	pop 100, swap 300 to 200
1000	push 1000	swap 200 to 1000	No operation	pop 100, swap 300 to 1000
vlan-tags outer 2000 inner 300	push 2000, push 300	swap 200 to 300, push 2000	push 2000	swap 100 to 2000
vlan-tags outer 100 inner 400	push 100, push 400	swap 200 to 400, push 100	push 100	swap 300 to 400
vlan-id-range 10-100	–	–	No operation	–
vlan-tags outer 200 inner-range 10-100	–	–	push 200	–

- Related Documentation**
- [Configuring Integrated Routing and Bridging for Bridge Domains on page 24](#)
  - [Layer 2 Learning and Forwarding Overview on page 7](#)
  - [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
  - [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)

## Configuring Integrated Routing and Bridging for Bridge Domains

Integrated routing and bridging (IRB) provides simultaneous support for Layer 2 bridging and Layer 3 routing on the same interface. IRB enables you to route packets to another routed interface or to another bridge domain that has an IRB interface configured. You configure a logical routing interface by including the `irb` statement at the `[edit interfaces]` hierarchy level and include that interface in the bridge domain. For more information about how to configure a routing interface, see the *Junos OS Network Interfaces Library for Routing Devices*.





**NOTE:** You can include only one routing interface in a bridge domain.

To configure a bridge domain with IRB support, include the following statements:

```
[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    interface interface-name;
    routing-interface routing-interface-name;
    service-id number;
    vlan-id (none | number);
    vlan-tags outer number inner number;
  }
}
```

For each bridge domain that you configure, specify a **bridge-domain-name**. You must also specify the value **bridge** for the **domain-type** statement.

For the **vlan-id** statement, you can specify either a valid VLAN identifier or the **none** option.



**NOTE:** If you configure a routing interface to support IRB in a bridge domain, you cannot use the **all** option for the **vlan-id** statement.

The **vlan-tags** statement enables you to specify a pair of VLAN identifiers; an **outer** tag and an **inner** tag.



**NOTE:** For a single bridge domain, you can include either the **vlan-id** statement or the **vlan-tags** statement, but not both.

For MC-LAG bridge domains, when the VLAN identifier is **none**, use the **service-id** statement to facilitate media access control (MAC) and Address Resolution Protocol (ARP) synchronization among MC-LAG peers.

To include one or more logical interfaces in the bridge domain, specify the interface name for each Ethernet interface to include that you configured at the **[edit interfaces]** hierarchy level.



**NOTE:** A maximum of 4000 active logical interfaces are supported on a bridge domain or on each mesh group in a VPLS routing instance configured for Layer 2 bridging.

To associate a routing interface with a bridge domain, include the **routing-interface routing-interface-name** statement and specify a **routing-interface-name** you configured at the **[edit interfaces irb]** hierarchy level. You can configure only one routing interface for each bridge domain. For more information about how to configure

logical and routing interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

In Junos OS Release 9.0 and later, IRB interfaces are supported for multicast snooping. For more information about multicast snooping, see the *Multicast Protocols Feature Guide for Routing Devices*.

In Junos 11.4 and later, IP multicast is supported on Layer 2 trunk ports through IRB interfaces using the Trio chipset.

In Junos OS Release 9.6 and later, in multihomed VPLS configurations, you can configure VPLS to keep a VPLS connection up if only an IRB interface is available by configuring the **irb** option for the **connectivity-type** statement at the **[edit routing-instances routing-instance-name protocols vpls]** hierarchy level. The **connectivity-type** statement has two options, **ce** and **irb**. The **ce** option is the default and specifies that a CE interface is required to maintain the VPLS connection. By default, if only an IRB interface is available, the VPLS connection is brought down. For more information about configuring VPNs, see the *Junos VPN Configuration Guide*.



**NOTE:** When you configure IRB interfaces in more than one logical system on a device, all of the of the IRB logical interfaces share the same MAC address.

On MX Series routers, although you can configure firewall filters and CoS rewrite rules on IRB interfaces, we recommend that you do not configure these functionalities on IRB interfaces because they do not work properly.

**Related Documentation**

- [Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 6](#)
- [Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19](#)

---

## Configuring Bridge Domains as Switches for Layer 2 Trunk Ports

You can configure a set of bridge domains that are associated with a Layer 2 trunk port. The set of bridge domains function as a switch. Packets received on a trunk interface are forwarded within a bridge domain that has the same VLAN identifier. A trunk interface also provides support for IRB, which provides support for Layer 2 bridging and Layer 3 IP routing on the same interface.

To configure a Layer 2 trunk port and set of bridge domains, include the following statements:

```
[edit interfaces]
interface-name {
  unit number {
    family bridge {
      interface-mode access;
      vlan-id number;
    }
  }
}
```

```

    }
  }
  interface-name {
    native-vlan-id number;
    unit number {
      family bridge {
        interface-mode trunk;
        vlan-id-list [ vlan-id-numbers ];
      }
    }
  }
}
[edit bridge-domains]
bridge-domain-name {
  vlan-id number;
  vlan-id-list [ vlan-id-numbers ];
  ....
}

```

For **interface-mode trunk**, you can include the **vlan-id-list** statement.

You must configure a bridge domain and VLAN identifier for each VLAN associated with the trunk interface. You can configure one or more trunk or access interfaces at the **[edit interfaces]** hierarchy level. An access interface enables you to accept packets with no VLAN identifier. For more information about configuring trunk and access interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

#### Related Documentation

- [Configuring a Bridge Domain on page 11](#)



## CHAPTER 4

# Layer 2 Virtual Switches

- [Configuring Integrated Routing and Bridging for a Bridge Domain in a Layer 2 Virtual Switch on page 29](#)
- [Configuring VPLS Ports in a Virtual Switch on page 30](#)
- [Configuring a Layer 2 Virtual Switch with a Layer 2 Trunk Port on page 32](#)
- [Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33](#)

## Configuring Integrated Routing and Bridging for a Bridge Domain in a Layer 2 Virtual Switch

---

Integrated routing and bridging (IRB) provides simultaneous support for Layer 2 bridging and Layer 3 IP routing on the same interface. IRB enables you to route local packets to another routed interface or to another bridge domain that has a Layer 3 protocol configured. You configure a logical routing interface by including the **irb** statement at **[edit interfaces]** hierarchy level and include that interface in the bridge domain. For more information about how to configure a routing interface, see the *Junos OS Network Interfaces Library for Routing Devices*.



**NOTE:** You can include only one routing interface in a bridge domain.

To configure a virtual switch with IRB support, include the following statements:

```
[edit]
routing-instances {
  routing-instance-name {
    instance-type virtual-switch;
    bridge-domains {
      bridge-domain-name {
        domain-type bridge;
        interface interface-name;
        routing-interface routing-interface-name;
        vlan-id (none | number);
        vlan-tags outer number inner number;
      }
    }
  }
}
```

To enable a virtual switch, you must specify **virtual-switch** as the **instance-type**. The **instance-type virtual-switch** statement is not supported at the **[edit logical-systems logical-system-name]** hierarchy level.

For each bridge domain that you configure for the virtual switch, specify a **bridge-domain-name**. You must also specify the value **bridge** for the **domain-type** statement.

For the **vlan-id** statement, you can specify either a valid VLAN identifier or the **none** option.



**NOTE:** For a single bridge domain, you can include either the **vlan-id** statement or the **vlan-tags** statement, but not both.

To include one or more logical interfaces in the bridge domain, specify the **interface-name** for each Ethernet interface to include that you configured at the **[edit interfaces irb]** hierarchy level.

To associate a routing interface with a bridge domain, include the **routing-interface routing-interface-name** statement and specify a **routing-interface-name** you configured at the **[edit interfaces irb]** hierarchy level. You can configure only one routing interface for each bridge domain. For more information about how to configure logical and routing interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.



**NOTE:** If you configure a routing interface to support IRB in a bridge domain, you cannot use the **all** option for the **vlan-id** statement.

**Related Documentation**

- [Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 6](#)
- [Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19](#)

---

## Configuring VPLS Ports in a Virtual Switch

In Junos OS Release 9.3 and later, you can configure VPLS ports in a virtual switch so that the logical interfaces of the Layer 2 bridge domains in the virtual switch can handle VPLS routing instance traffic. VPLS configuration no longer requires a dedicated routing instance of type **vpls**. Packets received on a Layer 2 trunk interface are forwarded within a bridge domain that has the same VLAN identifier.

A trunk interface is implicitly associated with bridge domains based on VLAN membership. Whereas access interfaces can be part of one VLAN only, trunk interfaces multiplex traffic from multiple VLANs and usually interconnect switches. A Layer 2 trunk port also supports IRB.

To configure VPLS ports in a virtual switch, perform the following tasks:

1. To configure the Layer 2 trunk ports that you will associate with the bridge domains in the virtual switch, include the following statements in the configuration:

```
[edit]
interfaces {
  interface-name {
    unit logical-unit-number { # Call this 'L2-trunk-port-A'
      family bridge {
        interface-mode trunk;
        vlan-id-list [ vlan-id-numbers ] ; # Trunk mode VLAN membership for this
        interface
      }
    }
  }
  .
  .
  .
  interface-name {
    unit logical-unit-number { # Call this 'L2-trunk-port-B'
      family bridge {
        interface-mode trunk;
        vlan-id-list [ vlan-id-numbers ] ; # Trunk mode VLAN membership for this
        interface
      }
    }
  }
}
```

To configure a logical interface as a trunk port, include the **interface-mode** statement and the **trunk** option at the **[edit interfaces interface-name unit logical-unit-number family bridge]** hierarchy level.

To configure all the VLAN identifiers to associate with a Layer 2 trunk port, include the **vlan-id-list [ vlan-id-numbers ]** statement at the **[edit interfaces interface-name unit logical-unit-number family bridge]** hierarchy level.

Each of the logical interfaces “**L2-trunk-port-A**” and “**L2-trunk-port-B**” accepts packets tagged with any VLAN ID specified in the respective **vlan-id-list** statements.

2. To configure a virtual switch consisting of a set of bridge domains that are associated with one or more logical interfaces configured as a trunk ports, include the following statements in the configuration:

```
[edit]
routing-instance {
  routing-instance-name
  instance-type virtual-switch;
  interface L2-trunk-port-A; # Include one trunk port
  interface L2-trunk-port-B; # Include the other trunk port
  bridge-domains {
    bridge-domain-name-0 {
      domain-type bridge;
      vlan-id number;
    }
    bridge-domain-name-1 {
```

```
        domain-type bridge;
        vlan-id number;
    }
}
protocols {
    vpls {
        vpls-id number;
        ... vpls-configuration ...
    }
}
}
```

To begin configuring a virtual switch, include the **instance-type** statement and the **virtual-switch** option at the **[edit routing-instances *routing-instance-name*]** hierarchy level.

To configure a virtual switch consisting of a set of bridge domains that are associated with one or more logical interfaces configured as a trunk ports, you must identify each logical interface by including the **interface *interface-name*** statement at the **[edit routing-instances *routing-instance-name*]** hierarchy level.

For each VLAN configured for a trunk port, you must configure a bridge-domain that includes the trunk port logical interface and uses a VLAN identifier within the range carried by that trunk interface. To configure, include the **domain-type bridge**, **vlan-id *number***, and statements at the **[edit routing-instances *routing-instance-name* bridge-domain *bridge-domain-name*]** hierarchy level.

**Related Documentation**

- [Configuring a Bridge Domain on page 11](#)

---

## Configuring a Layer 2 Virtual Switch with a Layer 2 Trunk Port

---

You can associate one or more Layer 2 trunk interfaces with a virtual switch. A Layer 2 trunk interface enables you to configure a logical interface to represent multiple VLANs on the physical interface. Within the virtual switch, you configure a bridge domain and VLAN identifier for each VLAN identifier configured on the trunk interfaces. Packets received on a trunk interface are forwarded within a bridge domain that has the same VLAN identifier. Each virtual switch you configure operates independently and can participate in a different Layer 2 network.

A virtual switch configured with a Layer 2 trunk port also supports IRB within a bridge domain. IRB provides simultaneous support for Layer 2 bridging and Layer 3 IP routing on the same interface. Only an interface configured with the **interface-mode (access | trunk)** statement can be associated with a virtual switch. An access interface enables you to accept packets with no VLAN identifier. For more information about configuring trunk and access interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

In addition, you can configure Layer 2 learning and forwarding properties for the virtual switch.



To configure a virtual switch with a Layer 2 trunk interface, include the following statements:

```
[edit]
routing-instances {
  routing-instance-name {
    instance-type virtual-switch;
    interface interface-name;
    bridge-domains {
      bridge-domain-name {
        vlan-id number;
      }
    }
  }
}
```



**NOTE:** You must configure a bridge domain and VLAN identifier for each VLAN identifier configured for the trunk interface.

**Related  
Documentation**

- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)

## Configuring a Layer 2 Virtual Switch for MX Series Routers

A Layer 2 virtual switch, which isolates a LAN segment with its spanning-tree protocol instance and separates its VLAN ID space, filters and forwards traffic only at the data link layer. Layer 3 routing is not performed. Each bridge domain consists of a set of logical ports that participate in Layer 2 learning and forwarding. A virtual switch represents a Layer 2 network.

Two main types of interfaces are used in virtual switch hierarchies:

- Layer 2 logical interface—This type of interface uses the VLAN-ID as a virtual circuit identifier and the scope of the VLAN-ID is local to the interface port. This type of interface is often used in service-provider-centric applications.
- Access or trunk interface—This type of interface uses a VLAN-ID with global significance. The access or trunk interface is implicitly associated with bridge domains based on VLAN membership. Access or trunk interfaces are typically used in enterprise-centric applications.



**NOTE:** The difference between access interfaces and trunk interfaces is that access interfaces can be part of one VLAN only and the interface is normally attached to an end-user device (packets are implicitly associated with the configured VLAN). In contrast, trunk interfaces multiplex traffic from multiple VLANs and usually interconnect switches.

To configure a Layer 2 virtual switch, include the following statements:

```
[edit]
routing-instances {
  routing-instance-name (
    instance-type virtual-switch;
    bridge-domains {
      bridge-domain-name {
        domain-type bridge;
        interface interface-name;
        vlan-id (all | none | number); # Cannot be used with 'vlan-tags' statement
        vlan-id-list [ vlan-id-numbers ];
        vlan-tags outer number inner number; # Cannot be used with 'vlan-id' statement
      }
    }
  }
  protocols {
    mstp {
      ...mstp-configuration ...
    }
  }
}
```

To enable a virtual switch, you must specify **virtual-switch** as the **instance-type**.

For each bridge domain that you configure for the virtual switch, specify a **bridge-domain-name**. You must also specify the value **bridge** for the **domain-type** statement.

For the **vlan-id** statement, you can specify either a valid VLAN identifier or the **none** or **all** options. If you specify a valid VLAN identifier, you cannot also use the **none** option. These statements are mutually exclusive.

The **all** option is not supported with IRB.



**NOTE:** You do not have to specify a VLAN identifier for a bridge domain. However, you cannot specify the same VLAN identifier for more than one bridge domain within a virtual switch. Each bridge domain within a virtual switch must have a unique VLAN identifier.



**NOTE:** For a single bridge domain, you can include either the **vlan-id** statement or the **vlan-tags** statement, but not both.

To specify one or more logical interfaces to include in the bridge domain, specify an **interface-name** for an Ethernet interface you configured at the **[edit interfaces]** hierarchy level. For more information, see the *Junos OS Network Interfaces Library for Routing Devices*.

#### Related Documentation

- [Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 6](#)
- [Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19](#)

- [Configuring Integrated Routing and Bridging for a Bridge Domain in a Layer 2 Virtual Switch on page 29](#)



## CHAPTER 5

# Layer 2 Learning and Forwarding for Bridge Domains

- [Disabling MAC Learning for a Bridge Domain or Logical Interface on page 37](#)
- [Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 38](#)
- [Configuring the Size of the MAC Address Table on page 39](#)
- [Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 40](#)
- [Enabling MAC Accounting for a Bridge Domain on page 41](#)

### Disabling MAC Learning for a Bridge Domain or Logical Interface

---

You can disable MAC learning for all logical interfaces in a specified bridge domain, or for a specific logical interface in a bridge domain. Disabling dynamic MAC learning prevents the specified interfaces from learning source MAC addresses.

To disable MAC learning for all logical interfaces in a bridge domain in a virtual switch, include the **no-mac-learning** statement at the **[edit bridge-domains *bridge-domain-name* bridge-options]** hierarchy level:

```
[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    interface interface-name;
    bridge-options {
      no-mac-learning;
    }
  }
}
```

To disable MAC learning for a specific logical interface in a bridge domain, include the **no-mac-learning** statement at the **[edit bridge-domains *bridge-domain-name* bridge-options interface *interface-name*]** hierarchy level.

```
[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    interface interface-name;
    bridge-options {
```

```

interface interface-name {
    no-mac-learning;
}
}
}

```



**NOTE:** When you disable MAC learning, source MAC addresses are not dynamically learned, and any packets sent to these source addresses are flooded into the bridge domain.



**NOTE:** When you gather interfaces into a bridge domain, the `no-mac-learn-enable` statement at the `[edit interfaces interface-name gigether-options ethernet-switch-profile]` hierarchy level is not supported. You must use the `no-mac-learning` statement at the `[edit bridge-domains bridge-domain-name bridge-options interface interface-name]` hierarchy level to disable MAC learning on an interface in a bridge domain.



**NOTE:** When MAC learning is disabled for a VPLS routing instance, traffic is not load balanced and only one of the equal-cost next hops is used.

#### Related Documentation

- [Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 38](#)
- [Configuring the Size of the MAC Address Table on page 39](#)
- [Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 40](#)
- [Enabling MAC Accounting for a Bridge Domain on page 41](#)

## Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain

You can manually add static MAC entries for the logical interfaces in a bridge domain. You can specify one or more static MAC addresses for each logical interface.

To add a static MAC address for a logical interface in a bridge domain, include the `static-mac mac-address` statement at the `[edit bridge-domains bridge-domain-name bridge-options interface interface-name]` hierarchy level.

```

[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    bridge-options {
      interface interface-name {
        static-mac mac-address {
          <vlan-id number>;
        }
      }
    }
  }
}

```

```

    }
  }
}

```

You can optionally specify a VLAN identifier for the static MAC address by using the **vlan-id** statement. To specify a VLAN identifier for a static MAC address, you must use the **all** option when configuring a VLAN identifier for the bridge domain.



**NOTE:** If a static MAC address you configure for a logical interface appears on a different logical interface, packets sent to that interface are dropped.

#### Related Documentation

- [Disabling MAC Learning for a Bridge Domain or Logical Interface on page 37](#)
- [Configuring the Size of the MAC Address Table on page 39](#)
- [Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 40](#)
- [Enabling MAC Accounting for a Bridge Domain on page 41](#)

## Configuring the Size of the MAC Address Table

You can modify the size of the MAC address table for each bridge domain. The default table size is 5120 addresses. The minimum you can configure is 16 addresses, and the maximum is 1,048,575 addresses.

If the MAC table limit is reached, new addresses can no longer be added to the table. Unused MAC addresses are removed from the MAC address table automatically. This frees space in the table, allowing new entries to be added.

To modify the size of the MAC table, include the **mac-table-size limit** statement at the **[edit bridge-domains *bridge-domain-name* bridge-options]** hierarchy level:

```

[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    bridge-options {
      mac-table-size limit {
        packet-action drop;
      }
    }
  }
}

```

#### Related Documentation

- [Disabling MAC Learning for a Bridge Domain or Logical Interface on page 37](#)
- [Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 38](#)
- [Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 40](#)
- [Enabling MAC Accounting for a Bridge Domain on page 41](#)

## Limiting MAC Addresses Learned from an Interface in a Bridge Domain

---

You can configure a limit on the number of MAC addresses learned from a specific bridge domain or from a specific logical interface that belongs to a bridge domain.

To configure a limit for the number of MAC addresses learned from each logical interface in a bridge domain, include the **interface-mac-limit** *limit* statement at the **[edit bridge-domains *bridge-domain-name* bridge-options]** hierarchy level:

```
[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    interface interface-name;
    bridge-options {
      interface-mac-limit limit;
    }
  }
}
```

To limit the number of MAC addresses learned from a specific logical interface in a bridge domain or an entire bridge domain, include the **interface-mac-limit** *limit* statement at the **[edit bridge-domains *bridge-domain-name* bridge-options interface *interface-name*]** or **[edit bridge-domains *bridge-domain-name* bridge-options]** hierarchy level:

```
[edit]
bridge-domains {
  bridge-domain-name {
    domain-type bridge;
    interface interface-name;
    bridge-options {
      interface-mac-limit limit{
        packet-action drop;
      }
    }
  }
  interface interface-name {
    interface-mac-limit limit{
      packet-action drop;
    }
  }
}
```

For an access port, the default limit on the maximum number of MAC addresses that can be learned on an access port is 1024. Because an access port can be configured in only one bridge domain in a network topology, the default limit is 1024 addresses, which is same as the limit for MAC addresses learned on a logical interface in a bridge domain (configured by including the **interface-mac-limit** *limit* statement at the **[edit bridge-domains *bridge-domain-name* bridge-options interface *interface-name*]** or **[edit bridge-domains *bridge-domain-name* bridge-options]** hierarchy level.

For a trunk port, the default limit on the maximum number of MAC addresses that can be learned on a trunk port is 8192. Because a trunk port can be associated with multiple bridge domains, the default limit is the same as the limit for MAC addresses learned on a



logical interface in a virtual switch instance (configured by including the **interface-mac-limit limit** statement at the **[edit routing-instances routing-instance-name switch- options interface interface-name]** for a virtual switch instance).

The value you configure for a specific logical interface overrides any value you specify for the entire bridge domain at the **[edit bridge-domains bridge-domain-name bridge-options]** hierarchy level.

The default limit to the number of MAC addresses that can be learned on a logical interface is 1024. The range that you can configure for a specific logical interface is 1 through 131,071.

After the MAC address limit is reached, the default is for any incoming packets with a new source MAC address to be forwarded. You can specify that the packets be dropped by including the **packet-action drop** statement. To specify that packets be dropped for the entire bridge domain, include the **packet-action drop** statement at the **[edit bridge-domains bridge-domain-name bridge-options interface-mac-limit limit]** hierarchy level:

```
[edit bridge-domains bridge-domain-name bridge-options interface-mac-limit limit]
packet-action drop;
```

To specify that the packets be dropped for a specific logical interface in a bridge domain, include the **packet-action drop** statement at the **[edit bridge-domains bridge-domain-name bridge-options interface interface-name interface-mac-limit limit]** hierarchy level:

```
[edit bridge-domains bridge-domain-name bridge-options interface interface-name
 interface-mac-limit limit]
packet-action drop;
```

You can also configure a limit to the number of MAC addresses learned for an MX Series router.

#### Related Documentation

- [Disabling MAC Learning for a Bridge Domain or Logical Interface on page 37](#)
- [Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 38](#)
- [Configuring the Size of the MAC Address Table on page 39](#)
- [Enabling MAC Accounting for a Bridge Domain on page 41](#)

## Enabling MAC Accounting for a Bridge Domain

By default, MAC accounting is disabled. You can enable packet counting for a bridge domain. When you enable packet accounting, the Junos OS maintains packet counters for each MAC address learned on the interfaces in the bridge domain.

To enable MAC accounting for a bridge domain, include the **mac-statistics** statement at the **[edit bridge-domains bridge-domain-name bridge-options]** hierarchy level:

```
[edit bridge-domains bridge-domain-name bridge-options]
mac-statistics;
```

#### Related Documentation

- [Disabling MAC Learning for a Bridge Domain or Logical Interface on page 37](#)
- [Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 38](#)

- [Configuring the Size of the MAC Address Table on page 39](#)
- [Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 40](#)

## CHAPTER 6

# Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports

- [Disabling MAC Learning for a Set of Bridge Domains on page 43](#)
- [Limiting the Number of MAC Addresses Learned from a Trunk Port on page 43](#)
- [Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 44](#)
- [Enabling MAC Accounting for a Set of Bridge Domains on page 45](#)

### Disabling MAC Learning for a Set of Bridge Domains

You can disable MAC learning for a set of bridge domains. Disabling dynamic MAC learning prevents the Layer 2 trunk port associated with the set of bridge domains from learning source and destination MAC addresses. When you disable MAC learning, source MAC addresses are not dynamically learned, and any packets sent to these source addresses are flooded into the switch.

To disable MAC learning for a set of bridge domains, include the **no-mac-learning** statement at the **[edit switch-options]** hierarchy level:

```
[edit switch-options]  
no-mac-learning;
```

#### Related Documentation

- [Limiting the Number of MAC Addresses Learned from a Trunk Port on page 43](#)
- [Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 44](#)
- [Enabling MAC Accounting for a Set of Bridge Domains on page 45](#)

### Limiting the Number of MAC Addresses Learned from a Trunk Port

You can configure a limit on the number of MAC addresses learned from a trunk port or from a specific trunk or access interface.

To limit the number of MAC addresses learned through a trunk port associated with a set of bridge domains, include the **interface-mac-limit** *limit* statement at the **[edit switch-options]** hierarchy level:

```
[edit]
switch-options {
  interface-mac-limit limit;
}
```

To limit the number of MAC addresses learned from a specific logical interface configured as an access interface or a trunk interface, include the **interface-mac-limit** *limit* statement at the **[edit switch-options interface *interface-name*]** hierarchy level:

```
[edit]
switch-options {
  interface interface-name {
    interface-mac-limit limit;
  }
}
```

The default value for the number MAC addresses that can be learned from a logical interface is 1024. You can specify a limit either for a set of bridge domains or for a specific logical interface in the range from 1 through 131,071. The value you configure for a specific logical interface overrides any value you specify for the set of bridge domains.

After the specified MAC address limit is reached, the default is for any incoming packets with a new source MAC address to be forwarded. You can specify that the packets be dropped for the entire virtual switch after the MAC address limit is reached by including the **packet-action drop** statement at the **[edit switch-options interface-mac-limit *limit*]** hierarchy level:

```
[edit switch-options interface interface-name interface-mac-limit limit]
packet-action drop;
```

To specify that the packets be dropped from a specific logical interface in a set of bridge domains with a trunk port after the MAC address limit is reached, include the **packet-action drop** statement at the **[edit routing-instances *routing-instance-name* interface *interface-name* interface-mac-limit *limit*]** hierarchy level:

```
[edit routing-instances routing-instance-name interface interface-name interface-mac-limit
limit]
packet-action drop;
```

#### Related Documentation

- [Disabling MAC Learning for a Set of Bridge Domains on page 43](#)
- [Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 44](#)
- [Enabling MAC Accounting for a Set of Bridge Domains on page 45](#)

---

## Configuring the Size of the MAC Address Table for a Set of Bridge Domains

You can modify the size of the MAC address table for a set of bridge domains. The minimum you can configure is 16 addresses, and the maximum is 1,048,575 addresses. The default table size is 5120 addresses.

If the MAC table limit is reached, new addresses can no longer be added to the table. Unused MAC addresses are removed from the MAC address table automatically. This frees space in the table, allowing new entries to be added to the table.

To modify the size of the MAC table for a set of bridge domains, include the **mac-table-size** statement at the **[edit switch-options]** hierarchy level:

```
[edit switch-options]  
mac-table-size limit;
```

**Related  
Documentation**

- [Disabling MAC Learning for a Set of Bridge Domains on page 43](#)
- [Limiting the Number of MAC Addresses Learned from a Trunk Port on page 43](#)
- [Enabling MAC Accounting for a Set of Bridge Domains on page 45](#)

---

## Enabling MAC Accounting for a Set of Bridge Domains

---

By default, MAC accounting is disabled. You can enable packet counting for a set of bridge domains. After you enable packet accounting, the Junos OS maintains packet counters for each MAC address learned on the trunk port associated with the set of bridge domains.

To enable MAC accounting for a set of bridge domains, include the **mac-statistics** statement at the **[edit switch-options]** hierarchy level:

```
[edit switch-options]  
mac-statistics;
```

**Related  
Documentation**

- [Disabling MAC Learning for a Set of Bridge Domains on page 43](#)
- [Limiting the Number of MAC Addresses Learned from a Trunk Port on page 43](#)
- [Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 44](#)



## CHAPTER 7

# Layer 3 Tunnel Services Interfaces with a DPC

- [Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC on page 47](#)

### Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC

The MX Series routers support Dense Port Concentrators (DPCs) with built-in Ethernet ports, which do not support Tunnel Services PICs. To create tunnel interfaces on an MX Series router with a DPC, you configure the DPC and the corresponding Packet Forwarding Engine to use for tunneling services at the **[edit chassis]** hierarchy level. You also configure the amount of bandwidth reserved for tunnel services. The Junos OS creates tunnel interfaces on the Packet Forwarding Engine.

To create tunnel interfaces on MX Series routers, include the following statements at the **[edit chassis]** hierarchy level:

```
[edit chassis]
fpc slot-number {
  pic number {
    tunnel-services {
      bandwidth bandwidth-value;
    }
  }
}
```

Include the **fpc slot-number** statement to specify the slot number of the DPC. If two SCBs are installed, the range is 0 through . If three SCBs are installed, the range is 0 through 5 and 7 through .

Include the **pic number** statement to specify the number of the Packet Forwarding Engine on the DPC. The range is 0 through 3.

You can also specify the amount of bandwidth to allocate for tunnel traffic on each Packet Forwarding Engine by including the **bandwidth bandwidth-value** statement at the **[edit chassis fpc slot-number pic pic-number]** hierarchy level:

- **1g** indicates that 1 Gbps of bandwidth is reserved for tunnel traffic. Configure this option only for a Packet Forwarding Engine on a Gigabit Ethernet 40-port DPC.

- **10g** indicates that 10 Gbps of bandwidth is reserved for tunnel traffic. Configure this option only for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.
- **20g or 40g**—Configure 20 gigabits per second or 40 gigabits per second only on an MX Series router with the MPC3E and the 100-Gigabit CFP MIC.

If you specify a bandwidth that is not compatible with the type of DPC and Packet Forwarding Engine, tunnel services are not activated. For example, you cannot specify a bandwidth of 1 Gbps for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.

When you configure tunnel interfaces on the Packet Forwarding Engine of a 10-Gigabit Ethernet 4-port DPC, the Ethernet interfaces for that port are removed from service and are no longer visible in the command-line interface (CLI). The Packet Forwarding Engine of a 10-Gigabit Ethernet 4-port DPC supports either tunnel interfaces or Ethernet interfaces, but not both. Each port on the 10-Gigabit Ethernet 4-port DPC includes two LEDs, one for tunnel services and one for Ethernet services, to indicate which type of service is being used. On the Gigabit Ethernet 40-port DPC, you can configure both tunnel and Ethernet interfaces at the same time.

To verify that the tunnel interfaces have been created, issue the **show interfaces terse** operational mode command. For more information, see the [CLI Explorer](#). The bandwidth that you specify determines the port number of the tunnel interfaces that are created. When you specify a bandwidth of **1g**, the port number is always 10. When you specify any other bandwidth, the port number is always 0.

For additional information about tunnel services, see the “Tunnel Services” chapter in the *Junos OS Services Interfaces Library for Routing Devices*.

**Related  
Documentation**

- *MX Series Router Architecture*
- *MX Series Router Packet Forwarding and Data Flow*
- *Line Cards Supported on MX Series Routers*



## CHAPTER 8

# Layer 2 Address Learning and Forwarding

- [Configuring the MAC Table Timeout Interval on page 49](#)
- [Enabling MAC Accounting on page 50](#)
- [Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 51](#)
- [Disabling Layer 2 Learning and Forwarding on page 52](#)
- [Example: Loop Detection Using the MAC Move Approach on page 52](#)

### Configuring the MAC Table Timeout Interval

---

The MAC table aging process ensures that a router tracks only active MAC addresses on the network and is able to flush out address that are no longer used.

You can configure the MAC table aging time, the maximum time that an entry can remain in the MAC table before it “ages out,” on all bridge domains, one or all VPLS instances, or one or all Ethernet virtual private network (EVPNs) instances on the router. This configuration can influence efficiency of network resource use by affecting the amount of traffic that is flooded to all interfaces because when traffic is received for MAC addresses no longer in the Ethernet routing table, the router floods the traffic to all interfaces.

Depending on how long you want to keep a MAC address in a MAC table before it expires, you can either increase or decrease the aging timer. By default, the timeout interval for all entries in the MAC table is 300 seconds. You can modify the timeout interval for MAC table entries on an MX Series router. You cannot modify the timeout interval for a virtual switch.



**NOTE:** The timeout interval applies only to dynamically learned MAC addresses. This value does not apply to configured static MAC addresses, which never time out.

The range for **seconds** is from 10 through 1,000,0000.

You can modify the timeout interval for a router (at the global level) or on a per-domain basis (bridge domain).

- To modify the timeout interval for the MAC table for a router:

```
[edit protocols l2-learning]
user@host# set global-mac-table-aging-time time;
```

- To modify the timeout interval for a bridge domain:

```
[edit bridge-domain bridge-domain-name bridge-options];
user@host# set mac-table-aging-time time;
```

- To modify the timeout for a VPLS or an Ethernet virtual private network (EVPN) instance within a bridge domain:

```
[edit routing-instance routing-instance-name protocols vpls];
[edit routing-instance routing-instance-name protocols evpn];
user@host# set mac-table-aging-time time;
```

#### Related Documentation

- [Layer 2 Learning and Forwarding Overview on page 7](#)
- [Enabling MAC Accounting on page 50](#)
- [Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 51](#)
- [Disabling Layer 2 Learning and Forwarding on page 52](#)

---

## Enabling MAC Accounting

By default, MAC accounting is disabled. On MX Series routers, you can enable packet accounting either for the router as a whole or for a specific bridge domain. After you enable packet accounting, the Junos OS maintains packet counters for each MAC address learned.

To enable MAC accounting for an MX Series router, include the **global-mac-statistics** statement at the **[edit protocols l2-learning]** hierarchy level:

```
[edit protocols l2-learning]
global-mac-statistics;
```

#### Related Documentation

- [Layer 2 Learning and Forwarding Overview on page 7](#)
- [Configuring the MAC Table Timeout Interval on page 49](#)
- [Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 51](#)
- [Disabling Layer 2 Learning and Forwarding on page 52](#)

## Limiting the Number of MAC Addresses Learned from Each Logical Interface

You can configure a limit to the number of MAC addresses learned from the logical interfaces on an MX Series router.

To configure a limit to the total number of MAC addresses that can be learned from the logical interfaces, include the **global-mac-limit** *limit* statement at the **[edit protocols l2-learning]** hierarchy level:

```
[edit]
protocols {
  l2-learning {
    global-mac-limit limit;
  }
}
```

The default limit to the number of MAC addresses that can be learned the router as a whole is 393,215. The range that you can configure for the router as a whole is 20 through 1,048,575.

After the configured MAC address limit is reached, the default is for packets to be forwarded. You can specify that the packets be dropped by including the **packet-action drop** statement at the **[edit protocols l2-learning global-mac-limit]** hierarchy level:

```
[edit]
protocols {
  l2-learning {
    global-mac-limit limit {
      packet-action drop;
    }
  }
}
```

You can also configure a limit to the number of MAC address learned from all the interfaces in a bridge domain or from a specific logical interface only. For more information, see “[Layer 2 Learning and Forwarding for Bridge Domains Overview](#)” on page 5.



**NOTE:** On MX Series routers running Junos OS Release 8.4 and later, statistics for an aged destination MAC entry are not retained. In addition, source and destination statistics are reset during a MAC move. In previous releases, only source statistics were reset during a MAC move.

### Related Documentation

- [Layer 2 Learning and Forwarding Overview on page 7](#)
- [Configuring the MAC Table Timeout Interval on page 49](#)
- [Enabling MAC Accounting on page 50](#)
- [Disabling Layer 2 Learning and Forwarding on page 52](#)

## Disabling Layer 2 Learning and Forwarding

---

Disabling dynamic MAC learning on an MX Series router or an EX Series switch prevents all the logical interfaces on the router or switch from learning source and destination MAC addresses.

To disable MAC learning for an MX Series router or an EX Series switch, include the **global-no-mac-learning** statement at the **[edit protocols l2-learning]** hierarchy level:

```
[edit protocols l2-learning]  
global-no-mac-learning;
```

For information about how to configure a virtual switch, see “Configuring a Layer 2 Virtual Switch for MX Series Routers” on page 33.

### Related Documentation

- [Layer 2 Learning and Forwarding Overview on page 7](#)
- [Configuring the MAC Table Timeout Interval on page 49](#)
- [Enabling MAC Accounting on page 50](#)
- [Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 51](#)

## Example: Loop Detection Using the MAC Move Approach

---

This example shows how to detect loops using the MAC move approach.

- [Requirements on page 52](#)
- [Overview on page 52](#)
- [Configuration on page 53](#)
- [Verification on page 55](#)

### Requirements

This example requires the following hardware and software components:

- MX Series 3D Universal Edge Routers
- Junos OS Release 13.2 running on all the devices

### Overview

When a MAC address appears on a different physical interface or within a different unit of the same physical interface and if this behavior occurs frequently, it is considered a MAC move.

Configuration errors at the network can force traffic into never ending circular paths. Once there are loops in the Layer 2 network, one of the symptoms is frequent MAC moves, which can be used for rectification of the problem. When it is observed that a source MAC address is moving among the ports, interface is blocked based on the configured **action-priority** for the interface. If the **action-priority** value configured for interfaces is the

same, the last interface for the bridge domain on which the MAC address move occurred is blocked.

## Configuration

**CLI Quick Configuration** To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

```
set interfaces ge-1/0/4 vlan-tagging
set interfaces ge-1/0/4 encapsulation flexible-ethernet-services
set interfaces ge-1/0/4 unit 10 encapsulation vlan-bridge
set interfaces ge-1/0/4 unit 10 vlan-id 10
set interfaces ge-1/0/4 unit 11 encapsulation vlan-bridge
set interfaces ge-1/0/4 unit 11 vlan-id 11
set interfaces ge-1/0/5 unit 0 family bridge interface-mode trunk
set interfaces ge-1/0/5 unit 0 family bridge vlan-id-list 10-12
set interfaces ge-1/0/6 unit 0 family bridge interface-mode trunk
set interfaces ge-1/0/6 unit 0 family bridge vlan-id-list 10-12
set bridge-domains bd10 vlan-id 10
set bridge-domains bd10 enable-mac-move-action
set bridge-domains bd10 bridge-options interface ge-1/0/5.0 action-priority 1
set bridge-domains bd10 bridge-options interface ge-1/0/6.0 action-priority 5
set bridge-domains bd11 vlan-id 11
set bridge-domains bd11 enable-mac-move-action
set bridge-domains bd12 vlan-id 12
```

In the previous example, all the interfaces, including the trunk interfaces in bd10 and bd11 will be monitored. If there are frequent MAC moves detected within interfaces ge-1/0/5 and ge-1/0/6, interface ge-1/0/5 is blocked. The blocking for trunk interfaces is such that data traffic only for a VLAN (on which the MAC move is detected) will be blocked and not for all the VLANs in the trunk. No action will be taken if a frequent MAC move is observed in bd12.

## Configuring Loop Detection Using the MAC Move Approach

**Step-by-Step Procedure** The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode*.

To configure loop detection using the MAC address move approach:

1. Configure the interfaces.

```
[edit interfaces]
user@host# set ge-1/0/4 vlan-tagging
user@host# set ge-1/0/4 encapsulation flexible-ethernet-services
user@host# set ge-1/0/4 unit 10 encapsulation vlan-bridge
user@host# set ge-1/0/4 unit 10 vlan-id 10
user@host# set ge-1/0/4 unit 11 encapsulation vlan-bridge
user@host# set ge-1/0/4 unit 11 vlan-id 11

user@host# set ge-1/0/5 unit 0 family bridge interface-mode trunk
```

```
user@host# set ge-1/0/5 unit 0 family bridge vlan-id-list 10-12
```

```
user@host# set ge-1/0/6 unit 0 family bridge interface-mode trunk
user@host# set ge-1/0/6 unit 0 family bridge vlan-id-list 10-12
```

2. Configure the bridge domain parameters.

```
[edit bridge-domains]
user@host# set bd10 vlan-id 10
user@host# set bd10 enable-mac-move-action
user@host# set bd10 bridge-options interface ge-1/0/5.0 action-priority 1
user@host# set bd10 bridge-options interface ge-1/0/6.0 action-priority 5
```

```
user@host# set bd11 vlan-id 11
user@host# set bd11 enable-mac-move-action
```

```
user@host# set bd12 vlan-id 12
```

---

## Results

From configuration mode, confirm your configuration by entering **show interfaces** and **show bridge-domains** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show interfaces
ge-1/0/4 {
  vlan-tagging;
  encapsulation flexible-ethernet-services;
  unit 10 {
    encapsulation vlan-bridge;
    vlan-id 10;
  }
  unit 11 {
    encapsulation vlan-bridge;
    vlan-id 11;
  }
}
ge-1/0/5 {
  unit 0 {
    family bridge {
      interface-mode trunk;
      vlan-id-list 10-12;
    }
  }
}
ge-1/0/6 {
  unit 0 {
    family bridge {
      interface-mode trunk;
      vlan-id-list 10-12;
    }
  }
}
```

```

user@host# show bridge-domains
bridge-domains {
  bd10 {
    vlan-id 10;
    bridge-options {
      interface ge-1/0/5.0 {
        action-priority 1;
      }
      interface ge-1/0/6.0 {
        action-priority 5
      }
    }
    enable-mac-move-action;
  }
  bd11 {
    vlan-id 11;
    enable-mac-move-action;
  }
  bd12 {
    vlan-id 12;
  }
}

```

If you are done configuring the device, enter **commit** from configuration mode.

## Verification

### Verifying That the Logical Interfaces Blocked Due to MAC Move Are Displayed

**Purpose** Ensure that the current set of logical interfaces blocked due to a MAC move, if any, are displayed.

**Action** From operational mode, enter the **show l2-learning mac-move-buffer active** command.

```

user@host# show l2-learning mac-move-buffer active
MAC Address: 00:00:00:00:01:01, VLAN Id: 0
  Time Rec : 2012-06-25 06:23:41   Bridge Domain: bd10
  Prev IFL : ge-1/0/5.0           New IFL: ge-1/0/6.0
  IFBD      : ge-1/0/6.0:10       Blocked   : YES

```

**Meaning** As a result of MAC move detection, one of the involved interface bridge domains will be blocked. The output shows that the ge-1/0/6 logical interface is blocked.

**Related Documentation**

- [bridge-domains on page 61](#)
- [Layer 2 Learning and Forwarding Overview on page 7](#)





## CHAPTER 9

# Configuration Statements for Bridge Domains

- [\[edit bridge-domains\] Hierarchy Level on page 57](#)
- [action-priority on page 60](#)
- [bridge-domains on page 61](#)
- [bridge-options on page 62](#)
- [domain-type on page 63](#)
- [enable-mac-move-action on page 63](#)
- [interface on page 64](#)
- [interface-mac-limit on page 65](#)
- [mac-statistics on page 67](#)
- [mac-table-size on page 69](#)
- [mac-table-aging-time on page 71](#)
- [no-irb-layer-2-copy on page 72](#)
- [no-mac-learning on page 73](#)
- [packet-action on page 75](#)
- [routing-interface on page 78](#)
- [service-id on page 79](#)
- [static-mac on page 80](#)
- [vlan-id \(Bridge Domain or VLAN\) on page 81](#)
- [vlan-id-list on page 83](#)
- [vlan-tags on page 84](#)

### [\[edit bridge-domains\] Hierarchy Level](#)

---

```
bridge-domains {  
  bridge-domain-name {  
    bridge-options {  
      ... the bridge-options subhierarchy appears after the main [edit bridge-domains  
        bridge-domain-name] hierarchy ...  
    }  
  }  
}
```

```

description text-description;
domain-type bridge;
forwarding-options {
  dhcp-relay {
    ... same statements as in [edit forwarding-options dhcp-relay] Hierarchy Level ...
  }
  filter {
    input filter-name;
  }
  flood {
    input filter-name;
  }
}
interface interface-name;
multicast-snooping-options {
  ... same statements as in [edit multicast-snooping-options] Hierarchy Level ...
}
no-irb-layer-2-copy;
no-local-switching;
protocols {
  ... the protocols subhierarchy appears after the main [edit bridge-domains
    bridge-domain-name] hierarchy ...
}
routing-interface irb-interface-name;
vlan-id (all | none | number);
service-id
enable-mac-move-action ;
vlan-id-list [ vlan-id-numbers ];
vlan-tags outer <tpid.>vlan-id <inner <tpid.>vlan-id>;
}

bridge-domain-name {
  bridge-options {
    interface interface-name {
      action-priority number;
      interface-mac-limit {
        limit;
        packet-action drop;
      }
      no-mac-learning;
      static-mac mac-address {
        vlan-id number;
      }
    }
  }
  interface-mac-limit {
    limit;
    packet-action drop;
  }
  mac-statistics;
  mac-table-size {
    number-of-addresses;
    packet-action drop;
  }
  no-mac-learning;
}
}

```

```

bridge-domain-name {
  protocols {
    igmp-snooping {
      immediate-leave;
      interface interface-name {
        group-limit number;
        host-only-interface;
        immediate-leave;
        multicast-router-interface;
        static {
          group multicast-ip-address {
            source multicast-ip-address;
          }
        }
      }
    }
  }
  proxy {
    source-address ip-address;
  }
  query-interval seconds;
  query-last-member-interval seconds;
  query-response-interval seconds;
  robust-count number;
  traceoptions {
    file filename <files number> <size maximum-file-size> <world-readable |
      no-world-readable>;
    flag flag <flag-modifier> <disable>;
  }
  vlan vlan-id {
    ... same statements as at the [edit bridge-domains bridge-domain-name protocols
      igmp-snooping] hierarchy level that appears in this topic, EXCEPT FOR ...
    traceoptions {...} # NOT valid at this level
    vlan vlan-id {...} # NOT valid at this level
  }
}
}
}
}

```

#### Related Documentation

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## action-priority

---

<b>Syntax</b>	<code>action-priority <i>number</i>;</code>
<b>Hierarchy Level</b>	[edit <code>bridge-domains <i>bridge-domain-name</i></code> <code>bridge-options</code> interface <i>interface-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 13.2.
<b>Description</b>	Configure the action priority value for an interface in a bridge domain on MAC move detection. This priority value is used to determine which interface should be blocked when a throttled MAC move is detected between two interfaces. The priority value can be between 0 and 7 inclusive. A higher value means lower priority. For example, if a MAC address move occurs between two interfaces with the action priority value set to 5 and 6, the interface with value 5 as the action priority value is blocked.
<b>Default</b>	4
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring a Bridge Domain on page 11</a></li><li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a></li></ul>

## bridge-domains

<b>Syntax</b>	<pre> bridge-domains <i>bridge-domain-name</i>;   bridge-options {     interface <i>interface-name</i> {       static-mac <i>mac-address</i>;       action-priority <i>number</i>;     }     interface-mac-limit <i>limit</i>;     mac-statistics;     mac-table-size <i>limit</i>;     no-mac-learning;   }   community-vlans <i>number number</i>;   isolated-vlan <i>vlan-id</i>   description;   domain-type bridge;   enable-mac-move-action;   forwarding-options   interface <i>interface-name</i>;   mcae-mac-flush;   mcae-mac-synchronize;   multicast-snooping-options;   no-irb-layer-2-copy;   no-local-switching;   protocols   routing-interface <i>routing-interface-name</i>;   service-id <i>number</i>;   vlan-id (all   none   <i>number</i>);   vlan-id-list [ <i>vlan-id-numbers</i> ];   vlan-tags outer <i>number</i> inner <i>number</i>; </pre>
<b>Hierarchy Level</b>	<pre> [edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i>], [edit routing-instances <i>routing-instance-name</i>] </pre>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>Support for the <b>no-irb-layer-2-copy</b> statement added in Junos OS Release 10.2.</p> <p>Support for <b>enable-mac-move-action</b> and <b>action-priority</b> added in Junos OS Release 13.2.</p> <p>Support for <b>service-id</b> statement added in Junos OS Release 13.2.</p> <p>Support for <b>forwarding-options</b> statement added in Junos OS Release 14.1.</p> <p>Support for <b>community-vlans</b> and <b>isolated-vlan</b> statements added in Junos OS Release 14.2 for MX240, MX480, and MX960 routers in enhanced LAN mode.</p>
<b>Description</b>	(MX Series routers only) Configure a domain that includes a set of logical ports that share the same flooding or broadcast characteristics in order to perform Layer 2 bridging.
<b>Options</b>	<i>bridge-domain-name</i> —Name of the bridge domain.



**NOTE:** Bridge domain names cannot include a forward slash (/).

The remaining statements are explained separately.

<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring a Bridge Domain on page 11</a></li><li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a></li></ul>

---

## bridge-options

---

<b>Syntax</b>	<pre>bridge-options {   interface <i>interface-name</i>;     static-mac <i>static-mac-address</i>;   }   interface-mac-limit <i>limit</i>;     packet-action drop;   }   mac-statistics;   mac-table-size <i>limit</i>;   mac-table-aging-time <i>time</i>;   no-mac-learning; }</pre>
<b>Hierarchy Level</b>	[edit <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
<b>Description</b>	(MX Series routers only) Configure Layer 2 learning and forwarding properties for a bridge domain or a virtual switch.  The remaining statements are explained separately.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li></ul>

## domain-type

<b>Syntax</b>	domain-type bridge;
<b>Hierarchy Level</b>	[edit <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit vlans <i>vlan-name</i> ] [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6. Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
<b>Description</b>	Define the type of domain for a Layer 2 bridge domain or VLAN.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring a Bridge Domain on page 11</a></li> <li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a></li> <li>• <a href="#">Configuring a Layer 2 Virtual Switch</a></li> </ul>

## enable-mac-move-action

<b>Syntax</b>	enable-mac-move-action;
<b>Hierarchy Level</b>	[edit <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 13.2.
<b>Description</b>	Enable the MAC move action feature at the bridge domain level. This statement blocks the logical interface for the bridge domain when a MAC move is detected on the interface.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring a Bridge Domain on page 11</a></li> </ul>

## interface

---

<b>Syntax</b>	<code>interface <i>interface-name</i>;</code>
<b>Hierarchy Level</b>	[edit <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit vlans <i>vlan-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.4. Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch. Support for logical systems added in Junos OS Release 9.6. Statement introduced in Junos OS Release 12.3R2 for EX Series switches. Statement introduced in Junos OS Release 15.1.
<b>Description</b>	(MX Series routers and EX Series switches only) Specify the logical interfaces to include in the bridge domain, VLAN, VPLS instance, or virtual switch.
<b>Options</b>	<i>interface-name</i> —Name of a logical interface. For more information about how to configure logical interfaces, see the <i>Junos OS Network Interfaces Library for Routing Devices</i> .
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring a Bridge Domain on page 11</a></li><li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a></li><li>• <i>Configuring a Layer 2 Virtual Switch</i></li><li>• <i>Tunnel Services Overview</i></li><li>• <i>Configuring Tunnel Interfaces on MX Series Routers</i></li></ul>



## interface-mac-limit

<b>Syntax</b>	<pre>interface-mac-limit <i>limit</i> {     packet-action drop; }</pre>
<b>Hierarchy Level</b>	<pre>[edit bridge-domains <i>bridge-domain-name</i> bridge-options], [edit bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options], [edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> switch-options], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> switch-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> switch-options], [edit logical-systems <i>logical-system-name</i> switch-options interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> switch-options], [edit routing-instances <i>routing-instance-name</i> switch-options interface <i>interface-name</i>], [edit switch-options], [edit switch-options interface <i>interface-name</i>], [edit switch-options interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> switch-options], [edit vlans <i>vlan-name</i> switch-options interface <i>interface-name</i>]</pre>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit switch-options], [edit switch-options interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> switch-options], and [edit vlans <i>vlan-name</i> switch-options interface <i>interface-name</i>] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	<p>Configure a limit to the number of MAC addresses that can be learned from a bridge domain, VLAN, virtual switch, or set of bridge domains or VLANs.</p>



**NOTE:** For multichassis link aggregation (MC-LAG) peers in active-active mode, configuring the `interface-mac-limit` statement or changing the `interface-mac-limit` configuration when traffic is flowing can cause the MAC entries to be out of synchronization between the two MC-LAG peers, which might result in flooding. To avoid flooding, you must either halt traffic forwarding and then configure the `interface-mac-limit` statement or use the `commit at configuration` statement to commit the changes at the same time in both the peer nodes.

Alternatively, if flooding does occur, you can clear the bridge MAC table on both the routers or switches by using the `clear bridge mac-table` command. Running this command ensures that the MAC entries are re-learned and in synchronization between both the peers.

---

<b>Default</b>	For an access port, the default MAC limit is 1024 MAC addresses. For a trunk port, the default MAC limit is 8192 MAC addresses.
<b>Options</b>	<b>limit</b> —Maximum number of MAC addresses learned from an interface. <b>Range:</b> 1 through 524287 MAC addresses per interface  The remaining statement is explained separately.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li><li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Overview</a></li><li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5</a></li><li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port</a></li></ul>

## mac-statistics

<b>Syntax</b>	mac-statistics;
<b>Hierarchy Level</b>	<p>[edit <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">switch-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> <a href="#">switch-options</a>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <a href="#">switch-options</a>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn],</p> <p>[edit <a href="#">switch-options</a>],</p> <p>[edit switch-options],</p> <p>[edit vlans <i>vlan-name</i> switch-options]</p>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit switch-options] and [edit vlans <i>vlan-name</i> switch-options] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Support for EVPNs added in Junos OS Release 13.2 for MX 3D Series routers.</p> <p>[edit switch-options] and [edit vlans <i>vlan-name</i> switch-options] hierarchy levels introduced in Junos OS Release 13.2 for the QFX Series.</p>
<b>Description</b>	(MX Series routers, EX Series switches, and QFX Series only) For bridge domains or VLANs, enable MAC accounting either for a specific bridge domain or VLAN, or for a set of bridge domains or VLANs associated with a Layer 2 trunk port.
<b>Default</b>	disabled
<b>Required Privilege Level</b>	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li> <li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Overview</a></li> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5</a></li> <li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port</a></li> <li>• <a href="#">Configuring EVPN Routing Instances</a></li> </ul>

- *Configuring EVPN Routing Instances on EX9200 Switches*

## mac-table-size

<b>Syntax</b>	<code>mac-table-size <i>limit</i> {     <b>packet-action</b> drop; }</code>
<b>Hierarchy Level</b>	<code>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</code> <code>[edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i>     <b>bridge-options</b>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i>     <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i>     <b>switch-options</b>],</code> <code>[edit logical-systems <i>logical-system-name</i> <b>switch-options</b>],</code> <code>[edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i>     <b>bridge-options</b>],</code> <code>[edit routing-instances <i>routing-instance-name</i> <b>switch-options</b>],</code> <code>[edit <b>switch-options</b>],</code> <code>[edit <b>switch-options</b>],</code> <code>[edit vlans <i>vlan-name</i> <b>switch-options</b>]</code>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p><b>[edit switch-options]</b> and <b>[edit vlans <i>vlan-name</i> switch-options]</b> hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Support at the <b>[edit vlans <i>vlan-name</i> switch-options]</b> hierarchy level introduced in Junos OS Release 13.2 for the QFX Series.</p>
<b>Description</b>	Modify the size of the MAC address table for the bridge domain or VLAN, a set of bridge domains or VLANs associated with a trunk port, or a virtual switch. The default is 5120 MAC addresses.



**NOTE:** For multichassis link aggregation (MC-LAG) peers in active-active mode, configuring the **mac-table-size** statement or changing the **mac-table-size** configuration when traffic is flowing can cause the MAC entries to be out of synchronization between the two MC-LAG peers, which might result in flooding. To avoid flooding, you must either halt traffic forwarding and then configure the **mac-table-size** statement or use the **commit at** configuration statement to commit the changes at the same time in both the peer nodes.

Alternatively, if flooding does occur, you can clear the bridge MAC table on both the routers by using the **clear bridge mac-table** command. Running this command ensures that the MAC entries are re-learned and in synchronization between both the peers.

**Options** *limit*—Specify the maximum number of addresses in the MAC address table.

**Range:** 16 through 1,048,575 MAC addresses

**Default:** 5120 MAC addresses There is no default MAC address limit for the **mac-table-size** statement at the **[edit switch-options]** hierarchy level. The number of MAC addresses that can be learned is only limited by the platform, 65,535 MAC addresses for EX Series switches and 1,048,575 MAC addresses for other devices.

The remaining statement is explained separately.



**Required Privilege** routing—To view this statement in the configuration.

**Level** routing-control—To add this statement to the configuration.

**Related  
Documentation**

- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- [Layer 2 Learning and Forwarding for VLANs Overview](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- [Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port](#)

## mac-table-aging-time

<b>Syntax</b>	<code>mac-table-aging-time <i>time</i>;</code>
<b>Hierarchy Level</b>	<p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vpls],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vpls]</p> <p>[edit <a href="#">bridge-options</a>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn]</p>
	<p> <b>NOTE:</b> For MX Series routers, the configuration statement is supported at the [bridge-options], [protocols vpls], and [protocols evpn] hierarchy levels only.</p>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 7.4.</p> <p>Statement introduced in Junos OS Release 15.1 for MX Series routers.</p>
<b>Description</b>	<p>Modify the timeout interval for the MAC table.</p> <p>For MX Series routers, you can use the <b>global-mac-table-aging-time</b> statement at the [edit protocols l2-learning] hierarchy level to configure the timeout interval at the global level or use the <b>mac-table-aging-time</b> to configure the timeout interval for a bridge domain or for a specific VPLS or EVPN instance. If multiple timeout interval values are configured on a router, the router determines the timeout interval value in the following order of priority:</p> <ul style="list-style-type: none"> <li>• Timeout interval configured at the VPLS or EVPN instance</li> <li>• Timeout interval configured for the bridge domain</li> <li>• Global timeout interval configured on the router</li> </ul> <p> <b>NOTE:</b> For MX Series routers, the timeout interval configuration feature is supported on routers with MPCs only.</p>
<b>Options</b>	<p><b>time</b>—Specify the number of seconds to wait between MAC table clearings.</p> <p><b>Range:</b> 10 through 1,000,000 seconds</p> <p><b>Default:</b> 300 seconds</p>
<b>Required Privilege Level</b>	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring VPLS Routing Instances</a></li> <li>• <a href="#">Configuring the MAC Table Timeout Interval on page 49</a></li> </ul>

## no-irb-layer-2-copy

---

<b>Syntax</b>	no-irb-layer-2-copy;
<b>Hierarchy Level</b>	[edit <a href="#">bridge-domains</a> ], [edit logical-routers <i>logical-router-name</i> <a href="#">bridge-domains</a> ], [edit routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.2.
<b>Description</b>	If you include this statement when using port mirroring with Integrated Routing and Bridging (IRB), then the packet is mirrored as a Layer 3 packet. By default, the packet is mirrored as a Layer 2 packet. This statement is also supported if a routing instance is set to type VPLS.
<b>Usage Guidelines</b>	See <a href="#">"Configuring a Bridge Domain" on page 11</a>
<b>Required Privilege Level</b>	view-level—To view this statement in the configuration. control-level—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a></li></ul>



## no-mac-learning

<b>Syntax</b>	no-mac-learning;
<b>Hierarchy Level</b>	<p>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>switch-options</b>],</p> <p>[edit logical-systems <i>logical-system-name</i> <b>switch-options</b>],</p> <p>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <b>switch-options</b>],</p> <p>[edit <b>switch-options</b>],</p> <p>[edit switch-options],</p> <p>[edit switch-options interface <i>interface-name</i>],</p> <p>[set vlans <i>vlan-name</i> switch-options]</p>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit <b>switch-options</b>], [edit <b>switch-options</b> interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> <b>switch-options</b>], and [edit vlans <i>vlan-name</i> <b>switch-options</b> interface <i>interface-name</i>] hierarchy levels introduced in Junos OS Release 12.3 R2 for EX Series switches.</p> <p>Support for EVPNs added in Junos OS Release 13.2 for MX 3D Series routers.</p> <p>Hierarchy levels [edit <b>switch-options</b> interface <i>interface-name</i>] and [edit vlans <i>vlan-name</i> <b>switch-options</b>] introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p>
<b>Description</b>	<p>For MX Series routers and EX Series switches, disable MAC learning for a virtual switch, for a bridge domain or VLAN, for a specific logical interface in a bridge domain or VLAN, or for a set of bridge domains or VLANs associated with a Layer 2 trunk port. On platforms that support EVPNs, you can disable MAC learning on an EVPN.</p>



**NOTE:** When MAC learning is disabled for a VPLS routing instance, traffic is not load-balanced and only one of the equal-cost next hops is used.

<b>Default</b>	MAC learning is enabled.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Configuring EVPN Routing Instances</i></li> <li>• <i>Configuring EVPN Routing Instances on EX9200 Switches</i></li> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li> <li>• <i>Layer 2 Learning and Forwarding for VLANs Overview</i></li> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5</a></li> <li>• <i>Understanding Bridging and VLANs on EX Series Switches</i></li> <li>• <i>Understanding Q-in-Q Tunneling on EX Series Switches</i></li> </ul>

## packet-action

**Syntax** `packet-action action;`

**Hierarchy Level**

```
[edit bridge-domains bridge-domain-name bridge-options interface interface-name
 interface-mac-limit limit],
[edit bridge-domains bridge-domain-name bridge-options interface-mac-limit limit],
[edit logical-systems logical-system-name bridge-domains bridge-domain-name
 bridge-options interface interface-name interface-mac-limit limit],
[edit logical-systems logical-system-name bridge-domains bridge-domain-name
 bridge-options interface-mac-limit limit],
[edit logical-systems logical-system-name routing-instances routing-instance-name
 bridge-domains bridge-domain-name bridge-options interface interface-name
 interface-mac-limit limit],
[edit logical-systems logical-system-name routing-instances routing-instance-name
 bridge-domains bridge-domain-name bridge-options interface-mac-limit limit],
[edit logical-systems logical-system-name routing-instances routing-instance-name
 switch-options interface interface-name interface-mac-limit limit],
[edit logical-systems logical-system-name routing-instances routing-instance-name
 switch-options interface-mac-limit limit],
[edit logical-systems logical-system-name switch-options interface-mac-limit limit],
[edit protocols l2-learning global-mac-limit limit],
[edit routing-instances routing-instance-name bridge-domains bridge-domain-name
 bridge-options interface interface-name interface-mac-limit limit],
[edit routing-instances routing-instance-name bridge-domains bridge-domain-name
 bridge-options interface-mac-limit limit],
[edit routing-instances routing-instance-name protocols evpn interface-mac-limit (VPLS)],
[edit routing-instances routing-instance-name protocols evpn interface interface-name
 interface-mac-limit (VPLS)],
[edit routing-instances routing-instance-name protocols evpn mac-table-size limit],
[edit routing-instances routing-instance-name switch-options interface interface-name
 interface-mac-limit limit],
[edit routing-instances routing-instance-name switch-options interface-mac-limit limit],
[edit switch-options interface interface-name interface-mac-limit limit],
[edit switch-options interface-mac-limit limit],
[edit switch-options interface interface-name interface-mac-limit limit],
[edit switch-options interface-mac-limit limit],
[edit switch-options mac-table-size limit],
[edit switch-options interface interface-name interface-mac-limit limit],
[edit vlans vlan-name switch-options interface interface-name interface-mac-limit limit],
[edit vlans vlan-name switch-options interface-mac-limit limit],
[edit vlans vlan-name switch-options mac-table-size limit],
[edit vlans vlan-name switch-options interface-mac-limit limit],
[edit vlans vlan-name switch-options interface interface-name interface-mac-limit limit],
[edit vlans vlan-name switch-options mac-table-size limit]
```

**Release Information**

Statement introduced in Junos OS Release 8.4.

Support for the **switch-options** statement added in Junos OS Release 9.2.

Support for top-level configuration for the **virtual-switch** type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.

Support for logical systems added in Junos OS Release 9.6.

[edit switch-options interface *interface-name* interface-mac-limit *limit*], [edit switch-options interface-mac-limit *limit*], [edit switch-options mac-table-size *limit*], [edit vlans *vlan-name* switch-options interface *interface-name* interface-mac-limit *limit*], [edit vlans *vlan-name* switch-options interface-mac-limit *limit*], and [edit vlans *vlan-name* switch-options mac-table-size *limit*] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.

Support for EVPNs introduced in Junos OS Release 13.2 on MX Series 3D Universal Edge Routers.

Support at the [edit switch-options interface *interface-name* interface-mac-limit *limit*] hierarchy level and hierarchy levels under [edit vlans *vlan-name*] introduced in Junos OS Release 13.2X50-D10 for EX Series switches and Junos OS Release 13.2 for the QFX Series.

**Description** Specify the action taken when packets with new source MAC addresses are received after the MAC address limit is reached. If this statement is not configured, packets with new source MAC addresses are forwarded by default.

**Default**



**NOTE:** On a QFX Series Virtual Chassis, if you include the shutdown option at the [edit vlans *vlan-name* switch-options interface *interface-name* interface-mac-limit packet-action] hierarchy level and issue the commit operation, the system generates a commit error. The system does not generate an error if you include the shutdown option at the [edit switch-options interface *interface-name* interface-mac-limit packet-action] hierarchy level.

Disabled. The default is for packets for new source MAC addresses to be forwarded after the MAC address limit is reached.

**Options**

- drop**—Drop packets with new source MAC addresses, and do not learn the new source MAC addresses.
- drop-and-log**—(EX Series switches and QFX Series only) Drop packets with new source MAC addresses, and generate an alarm, an SNMP trap, or a system log entry.
- log**—(EX Series switches and QFX Series only) Hold packets with new source MAC addresses, and generate an alarm, an SNMP trap, or a system log entry.
- none**—(EX Series switches and QFX Series only) Forward packets with new source MAC addresses, and learn the new source MAC address.
- shutdown**—(EX Series switches and QFX Series only) Disable the specified interface, and generate an alarm, an SNMP trap, or a system log entry.


**Required Privilege Level**

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related  
Documentation**


- *Configuring EVPN Routing Instances*
- *Configuring EVPN Routing Instances on EX9200 Switches*
- *Configuring MAC Limiting (CLI Procedure)*
- *Configuring Persistent MAC Learning (CLI Procedure)*
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- *Layer 2 Learning and Forwarding for VLANs Overview*
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- *Layer 2 Learning and Forwarding for VLANs Overview*
- *Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port*

## routing-interface

<b>Syntax</b>	<code>routing-interface <i>routing-interface-name</i>;</code>
<b>Hierarchy Level</b>	<code>[edit <b>bridge-domains</b> <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i></code> <code>    <b>bridge-domains</b> <i>bridge-domain-name</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i>]</code>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
<b>Description</b>	<p>(MX Series routers only) Specify a routing interface to include in a bridge domain or a VPLS routing instance.</p> <p>When you configure <b>routing-interface irb.x</b>, the VPLS connection comes up, even if no customer edge (CE) interfaces are configured. This works with one site configured, but not when multiple sites (multisite) are configured.</p>
<b>Options</b>	<p><b><i>routing-interface-name</i></b>—Name of the routing interface to include in the bridge domain or the VPLS routing instance. The format of the routing interface name is <b>irb.x</b>, where <b>x</b> is the unit number of the routing interface you configured at the <code>[edit interfaces irb]</code> hierarchy level. For more information about how to configure a routing interface, see the <i>Junos OS Network Interfaces Library for Routing Devices</i>.</p>
<div>  <p><b>NOTE:</b> You can specify only one routing interface for each bridge domain or VPLS instance.</p> </div>	
<b>Required Privilege Level</b>	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring a Bridge Domain on page 11</a></li> <li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a></li> </ul>

## service-id

---

<b>Syntax</b>	<code>service-id <i>number</i>;</code>
<b>Hierarchy Level</b>	<code>[edit bridge-domains <i>bridge-domain-name</i>]</code>
<b>Release Information</b>	Statement introduced in Junos OS Release 13.2
<b>Description</b>	Specify a service identifier to include in the packets sent to and from the multichassis link aggregation (MC-LAG) bridge domain when the VLAN identifier is set to <b>none</b> . This configuration facilitates media access control (MAC) and Address Resolution Protocol (ARP) synchronization among MC-LAG peers.
<div>  <b>NOTE:</b> The VLAN identifier none is supported only for IPv4 traffic. </div>	
<b>Options</b>	<b><i>number</i></b> —A valid service identifier. You must configure the same service identifier within the bridge domains of MC-LAG peers.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring a Bridge Domain on page 11</a></li> <li>• <a href="#">Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19</a></li> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li> <li>• <a href="#">bridge-domains on page 61</a></li> </ul>

## static-mac

---

Syntax	<pre>static-mac mac-address {     vlan-id number; }</pre>
Hierarchy Level	<pre>[edit bridge-domains bridge-domain-name bridge-options interface interface-name], [edit logical-systems logical-system-name bridge-domains bridge-domain-name     bridge-options interface interface-name], [edit logical-systems logical-system-name routing-instances routing-instance-name     bridge-domains bridge-domain-name bridge-options interface interface-name], [edit routing-instances routing-instance-name bridge-domains bridge-domain-name     bridge-options interface interface-name], [edit routing-instances routing-instance-name protocols evpn interface interface-name] [edit vlans vlan-name switch-options interface interface-name]</pre>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit vlans <i>vlan-name</i> switch-options interface <i>interface name</i>] hierarchy level introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Support for EVPNs added in Junos OS Release 13.2 for MX 3D Series routers. The <b>vlan-id</b> option is not available for EVPNs.</p> <p>[edit vlans <i>vlan-name</i> switch-options interface <i>interface name</i>] hierarchy level introduced in Junos OS Release 13.2 for the QFX Series.</p>
Description	<p>Configure a static MAC address for a logical interface in a bridge domain or VLAN.</p> <p>The <b>vlan-id</b> option can be specified for <b>static-macs</b> only if <b>vlan-id all</b> is configured for the bridging domain or VLAN.</p>
Options	<p><b>mac-address</b>—MAC address</p> <p><b>vlan-id number</b>—(Optional) VLAN identifier to associate with static MAC address.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"><li>• <i>Configuring EVPN Routing Instances</i></li><li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li><li>• <i>Layer 2 Learning and Forwarding for VLANs Overview</i></li></ul>



## vlan-id (Bridge Domain or VLAN)

<b>Syntax</b>	<code>vlan-id (all   none   <i>number</i>);</code>
<b>Hierarchy Level</b>	<code>[edit bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i></code> <code>bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]</code> <code>[edit vlans <i>vlan-name</i>]</code>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for Layer 2 trunk ports added in Junos OS Release 9.2.</p> <p>Support for SRX 3400, SRX 3600, SRX 5600, and SRX 5800 devices added in Junos OS Release 9.6.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	Specify a VLAN identifier (VID) to include in the packets sent to and from the bridge domain or VLAN, or a VPLS routing instance.



**NOTE:** When configuring a VLAN identifier for provider backbone bridge (PBB) routing instances, dual-tagged VIDs and the none option are not permitted.

**Options** *number*—A valid VLAN identifier. If you configure multiple bridge domains or VLANs with a valid VLAN identifier, you must specify a unique VLAN identifier for each. However, you can use the same VLAN identifier for bridge domains or VLANs that belong to different virtual switches. Use this option to send single tagged frames with the specified VLAN identifier over VPLS VT interfaces.



**NOTE:** If you specify a VLAN identifier, you cannot also use the all option. They are mutually exclusive.

**all**—Specify that the bridge domain or VLAN spans all the VLAN identifiers configured on the member logical interfaces.



**NOTE:** You cannot specify the all option if you include a routing interface in the bridge domain or VLAN.

**none**—Specify to enable shared VLAN learning or to send untagged frames over VPLS VT interfaces.




**NOTE:** Multichassis link aggregation (MC-LAG) supports the none option with the `vlan-id` statement with bridge domains or VLANs on Layer 2 networks and IPv4 networks. Starting with Junos OS Release 13.2R3, MC-LAG supports the none option with the `vlan-id` statement with bridge domains or VLANs on IPv6 networks.

---

<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
---------------------------------	---

<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring a Bridge Domain on page 11</a></li><li>• <a href="#">Configuring a VLAN</a></li><li>• <a href="#">Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19</a></li><li>• <a href="#">Configuring VLAN Identifiers for VLANs and VPLS Routing Instances</a></li><li>• <a href="#">Configuring Bridge Domains as Switches for Layer 2 Trunk Ports on page 26</a></li><li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a></li><li>• <a href="#">Configuring a Layer 2 Virtual Switch</a></li><li>• <a href="#">bridge-domains on page 61</a></li><li>• <a href="#">vlans</a></li></ul>
------------------------------	--

## vlan-id-list

<b>Syntax</b>	<code>vlan-id-list [ <i>vlan-id-numbers</i> ];</code>
<b>Hierarchy Level</b>	<p>[edit <b>bridge-domains</b> <i>bridge-domain-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i>]</p> <p>[edit interfaces <i>interface-name</i> unit 0],</p> <p>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>],</p> <p>[edit vlans <i>vlan-name</i>]</p>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 9.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Statement introduced in Junos OS Release 13.2 for the QFX Series.</p>
<b>Description</b>	<p>Specify a VLAN identifier list to use for a bridge domain or VLAN in trunk mode.</p> <p>Specify the <b>trunk</b> option in the <b>interface-mode</b> statement to accept packets with a VLAN ID that matches the list of VLAN IDs specified in the <b>vlan-id-list</b> statement to forward the packet within the bridge domain or VLAN configured with the matching VLAN ID. Specify the <b>access</b> option to accept packets with no VLAN ID to forward the packet within the bridge domain or VLAN configured with the VLAN ID that matches the VLAN ID specified in the <b>vlan-id</b> statement.</p> <p>This statement also enables you to bind a logical interface to a list of VLAN IDs, thereby configuring the logical interface to receive and forward a frame with a tag that matches the specified VLAN ID list.</p>
<b>Options</b>	<p><b><i>vlan-id-numbers</i></b>—Valid VLAN identifiers. You can combine individual numbers with range lists including a hyphen.</p> <p><b>Range:</b> 0 through 4095</p>
<div>  <p><b>NOTE:</b> On EX Series switches and the QFX Series, the range is 0 through 4094.</p> </div>	
<b>Required Privilege Level</b>	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring a Bridge Domain on page 11</a></li> <li>• <a href="#">Configuring a VLAN</a></li> <li>• <a href="#">Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19</a></li> <li>• <a href="#">Configuring VLAN Identifiers for VLANs and VPLS Routing Instances</a></li> </ul>

## vlan-tags

---

<b>Syntax</b>	<code>vlan-tags outer <i>number</i> inner <i>number</i>;</code>
<b>Hierarchy Level</b>	[edit <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit logical-systems <i>logical-system-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ], [edit routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> ] [edit vlans <i>vlan-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6. Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
<b>Description</b>	Specify dual VLAN identifier tags for a bridge domain, VLAN, or VPLS routing instance.
<b>Options</b>	<code>outer <i>number</i></code> —A valid VLAN identifier.  <code>inner <i>number</i></code> —A valid VLAN identifier.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring a Bridge Domain on page 11</a></li><li>• <a href="#">Configuring a VLAN</a></li><li>• <a href="#">Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 19</a></li><li>• <a href="#">Configuring VLAN Identifiers for VLANs and VPLS Routing Instances</a></li><li>• <a href="#">Configuring a Layer 2 Virtual Switch for MX Series Routers on page 33</a>.</li><li>• <a href="#">Configuring a Layer 2 Virtual Switch</a></li></ul>

## CHAPTER 10

# Configuration Statements for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports

- [\[edit switch-options\] Hierarchy Level on page 85](#)
- [switch-options on page 86](#)
- [interface-mac-limit on page 87](#)
- [mac-statistics on page 89](#)
- [mac-table-size on page 91](#)
- [no-mac-learning on page 93](#)
- [packet-action on page 95](#)

### [\[edit switch-options\] Hierarchy Level](#)

---

```
switch-options {  
  interface interface-name {  
    interface-mac-limit {  
      number-of-addresses;  
      packet-action drop;  
    }  
    no-mac-learning;  
  }  
  interface-mac-limit {  
    number-of-addresses;  
    packet-action drop;  
  }  
  mac-statistics;  
  mac-table-size {  
    number-of-addresses;  
    packet-action drop;  
  }  
  no-mac-learning;  
}
```

#### Related Documentation

- [Notational Conventions Used in Junos OS Configuration Hierarchies](#)

## switch-options

---

<b>Syntax</b>	<pre>switch-options {     interface <i>interface-name</i> {         interface-mac-limit <i>limit</i>;     }     interface-mac-limit <i>limit</i> {         packet-action drop;     }     mac-statistics;     mac-table-size <i>limit</i> {         packet-action drop;     }     no-mac-learning;     route-distinguisher (<i>as-number:id</i>   <i>ip-address:id</i>);     service-id <i>number</i>; <i>number</i>;     vrf-target {         community;         auto         import <i>community-name</i>;         export <i>community-name</i>;     }     vrf-import[ <i>policy-names</i> ];     vrf-export[ <i>policy-names</i> ]; }</pre>
<b>Hierarchy Level</b>	[edit], [edit logical-systems <i>logical-system-name</i> ], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> ], [edit routing-instances <i>routing-instance-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6. 14.1x53-D10
<b>Description</b>	Configure Layer 2 learning and forwarding properties for a set of bridge domains.
<b>Options</b>	The remaining statements are explained separately.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5</a></li></ul>

## interface-mac-limit

<b>Syntax</b>	<pre>interface-mac-limit <i>limit</i> {     <b>packet-action</b> drop; }</pre>
<b>Hierarchy Level</b>	<pre>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>switch-options</b>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>switch-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> <b>switch-options</b>], [edit logical-systems <i>logical-system-name</i> <b>switch-options</b> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> <b>switch-options</b>], [edit routing-instances <i>routing-instance-name</i> <b>switch-options</b> interface <i>interface-name</i>], [edit <b>switch-options</b>], [edit <b>switch-options</b> interface <i>interface-name</i>], [edit switch-options interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> switch-options], [edit vlans <i>vlan-name</i> switch-options interface <i>interface-name</i>]</pre>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit switch-options], [edit switch-options interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> switch-options], and [edit vlans <i>vlan-name</i> switch-options interface <i>interface-name</i>] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	<p>Configure a limit to the number of MAC addresses that can be learned from a bridge domain, VLAN, virtual switch, or set of bridge domains or VLANs.</p>



**NOTE:** For multichassis link aggregation (MC-LAG) peers in active-active mode, configuring the `interface-mac-limit` statement or changing the `interface-mac-limit` configuration when traffic is flowing can cause the MAC entries to be out of synchronization between the two MC-LAG peers, which might result in flooding. To avoid flooding, you must either halt traffic forwarding and then configure the `interface-mac-limit` statement or use the `commit at configuration` statement to commit the changes at the same time in both the peer nodes.

Alternatively, if flooding does occur, you can clear the bridge MAC table on both the routers or switches by using the `clear bridge mac-table` command. Running this command ensures that the MAC entries are re-learned and in synchronization between both the peers.

---

<b>Default</b>	For an access port, the default MAC limit is 1024 MAC addresses. For a trunk port, the default MAC limit is 8192 MAC addresses.
<b>Options</b>	<b>limit</b> —Maximum number of MAC addresses learned from an interface. <b>Range:</b> 1 through 524287 MAC addresses per interface  The remaining statement is explained separately.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li><li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Overview</a></li><li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5</a></li><li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port</a></li></ul>



## mac-statistics

<b>Syntax</b>	mac-statistics;
<b>Hierarchy Level</b>	<p>[edit <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <a href="#">switch-options</a>],</p> <p>[edit logical-systems <i>logical-system-name</i> <a href="#">switch-options</a>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <a href="#">bridge-domains</a> <i>bridge-domain-name</i> <a href="#">bridge-options</a>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <a href="#">switch-options</a>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn],</p> <p>[edit <a href="#">switch-options</a>],</p> <p>[edit switch-options],</p> <p>[edit vlans <i>vlan-name</i> switch-options]</p>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit switch-options] and [edit vlans <i>vlan-name</i> switch-options] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Support for EVPNs added in Junos OS Release 13.2 for MX 3D Series routers.</p> <p>[edit switch-options] and [edit vlans <i>vlan-name</i> switch-options] hierarchy levels introduced in Junos OS Release 13.2 for the QFX Series.</p>
<b>Description</b>	(MX Series routers, EX Series switches, and QFX Series only) For bridge domains or VLANs, enable MAC accounting either for a specific bridge domain or VLAN, or for a set of bridge domains or VLANs associated with a Layer 2 trunk port.
<b>Default</b>	disabled
<b>Required Privilege Level</b>	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li> <li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Overview</a></li> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5</a></li> <li>• <a href="#">Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port</a></li> <li>• <a href="#">Configuring EVPN Routing Instances</a></li> </ul>

- *Configuring EVPN Routing Instances on EX9200 Switches*

## mac-table-size

<b>Syntax</b>	<code>mac-table-size limit {     packet-action drop; }</code>
<b>Hierarchy Level</b>	<code>[edit bridge-domains bridge-domain-name bridge-options],</code> <code>[edit logical-systems logical-system-name bridge-domains bridge-domain-name bridge-options],</code> <code>[edit logical-systems logical-system-name routing-instances routing-instance-name bridge-domains bridge-domain-name bridge-options],</code> <code>[edit logical-systems logical-system-name routing-instances routing-instance-name switch-options],</code> <code>[edit logical-systems logical-system-name switch-options],</code> <code>[edit routing-instances routing-instance-name bridge-domains bridge-domain-name bridge-options],</code> <code>[edit routing-instances routing-instance-name switch-options],</code> <code>[edit switch-options],</code> <code>[edit vlans vlan-name switch-options]</code>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p><b>[edit switch-options]</b> and <b>[edit vlans vlan-name switch-options]</b> hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Support at the <b>[edit vlans vlan-name switch-options]</b> hierarchy level introduced in Junos OS Release 13.2 for the QFX Series.</p>
<b>Description</b>	Modify the size of the MAC address table for the bridge domain or VLAN, a set of bridge domains or VLANs associated with a trunk port, or a virtual switch. The default is 5120 MAC addresses.



**NOTE:** For multichassis link aggregation (MC-LAG) peers in active-active mode, configuring the **mac-table-size** statement or changing the **mac-table-size** configuration when traffic is flowing can cause the MAC entries to be out of synchronization between the two MC-LAG peers, which might result in flooding. To avoid flooding, you must either halt traffic forwarding and then configure the **mac-table-size** statement or use the **commit at** configuration statement to commit the changes at the same time in both the peer nodes.

Alternatively, if flooding does occur, you can clear the bridge MAC table on both the routers by using the **clear bridge mac-table** command. Running this command ensures that the MAC entries are re-learned and in synchronization between both the peers.

**Options** *limit*—Specify the maximum number of addresses in the MAC address table.

**Range:** 16 through 1,048,575 MAC addresses

**Default:** 5120 MAC addresses There is no default MAC address limit for the **mac-table-size** statement at the **[edit switch-options]** hierarchy level. The number of MAC addresses that can be learned is only limited by the platform, 65,535 MAC addresses for EX Series switches and 1,048,575 MAC addresses for other devices.

The remaining statement is explained separately.

**Required Privilege** routing—To view this statement in the configuration.

**Level** routing-control—To add this statement to the configuration.

**Related  
Documentation**

- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- [Layer 2 Learning and Forwarding for VLANs Overview](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- [Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port](#)

## no-mac-learning

<b>Syntax</b>	no-mac-learning;
<b>Hierarchy Level</b>	<p>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>switch-options</b>],</p> <p>[edit logical-systems <i>logical-system-name</i> <b>switch-options</b>],</p> <p>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> <b>switch-options</b>],</p> <p>[edit <b>switch-options</b>],</p> <p>[edit switch-options],</p> <p>[edit switch-options interface <i>interface-name</i>],</p> <p>[set vlans <i>vlan-name</i> switch-options]</p>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit <b>switch-options</b>], [edit <b>switch-options</b> interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> <b>switch-options</b>], and [edit vlans <i>vlan-name</i> <b>switch-options</b> interface <i>interface-name</i>] hierarchy levels introduced in Junos OS Release 12.3 R2 for EX Series switches.</p> <p>Support for EVPNs added in Junos OS Release 13.2 for MX 3D Series routers.</p> <p>Hierarchy levels [edit <b>switch-options</b> interface <i>interface-name</i>] and [edit vlans <i>vlan-name</i> <b>switch-options</b>] introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p>
<b>Description</b>	<p>For MX Series routers and EX Series switches, disable MAC learning for a virtual switch, for a bridge domain or VLAN, for a specific logical interface in a bridge domain or VLAN, or for a set of bridge domains or VLANs associated with a Layer 2 trunk port. On platforms that support EVPNs, you can disable MAC learning on an EVPN.</p>



**NOTE:** When MAC learning is disabled for a VPLS routing instance, traffic is not load-balanced and only one of the equal-cost next hops is used.

<b>Default</b>	MAC learning is enabled.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Configuring EVPN Routing Instances</i></li> <li>• <i>Configuring EVPN Routing Instances on EX9200 Switches</i></li> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5</a></li> <li>• <i>Layer 2 Learning and Forwarding for VLANs Overview</i></li> <li>• <a href="#">Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5</a></li> <li>• <i>Understanding Bridging and VLANs on EX Series Switches</i></li> <li>• <i>Understanding Q-in-Q Tunneling on EX Series Switches</i></li> </ul>

## packet-action

**Syntax** `packet-action action;`

**Hierarchy Level** [edit **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit **bridge-domains** *bridge-domain-name* **bridge-options** interface **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **switch-options** interface **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* **switch-options** interface **interface-mac-limit** *limit*],  
 [edit protocols **l2-learning** **global-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface **interface-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* protocols evpn interface-mac-limit (VPLS)],  
 [edit routing-instances *routing-instance-name* protocols evpn interface *interface-name* interface-mac-limit (VPLS)],  
 [edit routing-instances *routing-instance-name* protocols evpn mac-table-size *limit*],  
 [edit routing-instances *routing-instance-name* **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* **switch-options** interface **interface-mac-limit** *limit*],  
 [edit **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit switch-options **interface-mac-limit** *limit*],  
 [edit switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit switch-options **interface-mac-limit** *limit*],  
 [edit switch-options **mac-table-size** *limit*],  
 [edit switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options **mac-table-size** *limit*],  
 [edit vlans *vlan-name* switch-options **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options **mac-table-size** *limit*]

**Release Information** Statement introduced in Junos OS Release 8.4.  
 Support for the **switch-options** statement added in Junos OS Release 9.2.  
 Support for top-level configuration for the **virtual-switch** type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.

Support for logical systems added in Junos OS Release 9.6.

[edit switch-options interface *interface-name* interface-mac-limit *limit*], [edit switch-options interface-mac-limit *limit*], [edit switch-options mac-table-size *limit*], [edit vlans *vlan-name* switch-options interface *interface-name* interface-mac-limit *limit*], [edit vlans *vlan-name* switch-options interface-mac-limit *limit*], and [edit vlans *vlan-name* switch-options mac-table-size *limit*] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.

Support for EVPNs introduced in Junos OS Release 13.2 on MX Series 3D Universal Edge Routers.

Support at the [edit switch-options interface *interface-name* interface-mac-limit *limit*] hierarchy level and hierarchy levels under [edit vlans *vlan-name*] introduced in Junos OS Release 13.2X50-D10 for EX Series switches and Junos OS Release 13.2 for the QFX Series.

**Description** Specify the action taken when packets with new source MAC addresses are received after the MAC address limit is reached. If this statement is not configured, packets with new source MAC addresses are forwarded by default.

**Default**



**NOTE:** On a QFX Series Virtual Chassis, if you include the shutdown option at the [edit vlans *vlan-name* switch-options interface *interface-name* interface-mac-limit packet-action] hierarchy level and issue the commit operation, the system generates a commit error. The system does not generate an error if you include the shutdown option at the [edit switch-options interface *interface-name* interface-mac-limit packet-action] hierarchy level.

Disabled. The default is for packets for new source MAC addresses to be forwarded after the MAC address limit is reached.

**Options**

- drop**—Drop packets with new source MAC addresses, and do not learn the new source MAC addresses.
- drop-and-log**—(EX Series switches and QFX Series only) Drop packets with new source MAC addresses, and generate an alarm, an SNMP trap, or a system log entry.
- log**—(EX Series switches and QFX Series only) Hold packets with new source MAC addresses, and generate an alarm, an SNMP trap, or a system log entry.
- none**—(EX Series switches and QFX Series only) Forward packets with new source MAC addresses, and learn the new source MAC address.
- shutdown**—(EX Series switches and QFX Series only) Disable the specified interface, and generate an alarm, an SNMP trap, or a system log entry.

**Required Privilege Level**

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.



**Related  
Documentation**

- *Configuring EVPN Routing Instances*
- *Configuring EVPN Routing Instances on EX9200 Switches*
- *Configuring MAC Limiting (CLI Procedure)*
- *Configuring Persistent MAC Learning (CLI Procedure)*
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- *Layer 2 Learning and Forwarding for VLANs Overview*
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- *Layer 2 Learning and Forwarding for VLANs Overview*
- *Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port*





## CHAPTER 11

# Configuration Statements for Layer 3 Tunnel Services Interfaces with a DPC

- `tunnel-services` (Chassis) on page 100
- `bandwidth` (Tunnel Services) on page 102

## tunnel-services (Chassis)

<b>Syntax</b>	<pre>tunnel-services {     bandwidth (1g   10g   20g   40g);     tunnel-only; }</pre>
<b>Hierarchy Level</b>	[edit chassis fpc <i>slot-number</i> pic <i>number</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2.
<b>Description</b>	<p>For MX Series 3D Universal Edge Routers, configure the amount of bandwidth for tunnel services.</p> <p>For M7i, M10i, M120, M320, T Series and TX Matrix routers with IQ2 PICs and IQ2E PICs, configure support for per unit scheduling for GRE tunnels. You can specify the IQ2 and IQ2E PICs to work exclusively in tunnel mode or as a regular PIC. The default setting uses IQ2 and IQ2E PICs as a regular PIC. If you do not configure the <b>tunnel-only</b> option, the IQ2 and IQ2E PICs operate as regular PICs. For M7i, M10i, M120, M320, T Series and TX Matrix routers with IQ2 PICs and IQ2E PICs, you can use the <b>tunnel-only</b> option to specify that an IQ2 or IQ2E PIC work in tunnel mode only.</p>
	<p> <b>NOTE:</b> Bandwidth rates of 20 gigabits per second and 40 gigabits per second require use of an MX Series router with the 100-Gigabit Ethernet Modular Port Concentrator (MPC) and the 100-Gigabit CFP MIC.</p>
	<p> <b>NOTE:</b> On MX80 routers and MX Series routers with Trio-based FPCs, when ingress queuing is enabled for a PIC, tunnel services and inline services are not supported on the same PIC.</p>
<b>Options</b>	<p><b>tunnel-only</b> (Optional)—For M7i, M10i, M120, M320, T Series and TX Matrix routers with IQ2 PICs and IQ2E PICs, specify that an IQ2 or IQ2E PIC work in tunnel mode only.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Example: Configuring Tunnel Interfaces on a Gigabit Ethernet 40-Port DPC</i></li> <li>• <i>Example: Configuring Tunnel Interfaces on a 10-Gigabit Ethernet 4-Port DPC</i></li> <li>• <i>Example: Configuring Tunnel Interfaces on the MPC3E</i></li> <li>• <a href="#">bandwidth (Tunnel Services) on page 102</a></li> <li>• <i>[edit chassis] Hierarchy Level</i></li> </ul>

- [Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC on page 47](#)

## bandwidth (Tunnel Services)

<b>Syntax</b>	<code>bandwidth <i>bandwidth-value</i>;</code>
<b>Hierarchy Level</b>	[edit chassis fpc <i>slot-number</i> pic <i>number</i> <b>tunnel-services</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2.
<b>Description</b>	(MX Series 3D Universal Edge Routers and T4000 Core Routers only) Specify the amount of bandwidth in gigabits per second to reserve for tunnel services.
<b>Options</b>	<i>bandwidth-value</i> —Define the amount of bandwidth in gigabits per second to reserve for tunnel services. On MX Series routers, the bandwidth values can be <b>1g</b> , <b>10g</b> , <b>20g</b> , or <b>40g</b> . On T4000 routers, the bandwidth values are multiples of <b>10g</b> up to <b>100g</b> .



**NOTE:** The bandwidth that you specify determines the port number of the tunnel interfaces that are created. When you specify a bandwidth of **1g**, the port number is always 10. When you specify any other bandwidth, the port number is always 0.



**NOTE:** If you specify a bandwidth that is not compatible with the type of DPCs or MPCs and their respective Packet Forwarding Engine, tunnel services are not activated. For example, you cannot specify 1 gigabit per second bandwidth for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.



**NOTE:** Bandwidth rates of 20 gigabits per second and 40 gigabits per second require use of an MX Series router with the MPC3E and the 100-Gigabit CFP MIC.

<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
---------------------------------	---

<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Example: Configuring Tunnel Interfaces on a Gigabit Ethernet 40-Port DPC</i></li> <li>• <i>Configuring Tunnel Interfaces on MX Series Routers</i></li> <li>• <i>Configuring Tunnel Interfaces on T4000 Routers</i></li> <li>• <i>Example: Configuring Tunnel Interfaces on a 10-Gigabit Ethernet 4-Port DPC</i></li> <li>• <i>Example: Configuring Tunnel Interfaces on the MPC3E</i></li> </ul>
------------------------------	--

- [Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC on page 47](#)
- [tunnel-services \(Chassis\) on page 100](#)
- [\[edit chassis\] Hierarchy Level](#)





## CHAPTER 12

# Configuration Statements for Layer 2 Address Learning and Forwarding

- [\[edit protocols l2-learning\] Hierarchy Level](#) on page 105
- [l2-learning](#) on page 106
- [global-mac-limit](#) on page 107
- [global-mac-statistics](#) on page 107
- [global-mac-table-aging-time](#) on page 108
- [global-no-mac-learning](#) on page 108
- [interface-mac-limit](#) on page 109
- [packet-action](#) on page 111

### [\[edit protocols l2-learning\] Hierarchy Level](#)

---

```
protocols {  
  l2-learning {  
    global-mac-limit {  
      limit;  
      packet-action drop;  
    }  
    global-mac-statistics;  
    global-mac-table-aging-time seconds;  
    global-no-mac-learning;  
  }  
}
```

#### Related Documentation

- *Notational Conventions Used in Junos OS Configuration Hierarchies*
- [\[edit protocols\] Hierarchy Level](#)

## l2-learning

---

<b>Syntax</b>	<pre>l2-learning {     global-mac-limit <i>limit</i>;     global-mac-statistics;     global-mac-table-aging-time <i>seconds</i>;     global-no-mac-learning; }</pre>
<b>Hierarchy Level</b>	[edit protocols]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.4.
<b>Description</b>	<p>(MX Series routers, and EX Series switches, and QFX Series switches only) Configure Layer 2 address learning and forwarding properties globally.</p> <p>The statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Layer 2 Learning and Forwarding Overview on page 7</a></li></ul>

## global-mac-limit

<b>Syntax</b>	<code>global-mac-limit <i>limit</i> {     <code>packet-action</code> drop; }</code>
<b>Hierarchy Level</b>	[edit protocols <a href="#">l2-learning</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
<b>Description</b>	(MX Series routers and EX Series switches only) Limit the number of media access control (MAC) addresses learned from the logical interfaces on the router.
<b>Default</b>	393,215 MAC addresses
<b>Options</b>	<i>limit</i> —Number of MAC addresses that can be learned systemwide. <b>Range:</b> 20 through 1,048,575  The remaining statement is explained separately in the “Summary of Bridge Domain Configuration Statements” chapter.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 51</a></li> </ul>

## global-mac-statistics

<b>Syntax</b>	<code>global-mac-statistics;</code>
<b>Hierarchy Level</b>	[edit protocols <a href="#">l2-learning</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
<b>Description</b>	(MX Series routers and EX Series switches only) Enable MAC accounting for the entire router or switch.
<b>Default</b>	disabled
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Enabling MAC Accounting on page 50</a></li> </ul>

## global-mac-table-aging-time

---

<b>Syntax</b>	global-mac-table-aging-time <i>seconds</i> ;
<b>Hierarchy Level</b>	[edit protocols <a href="#">l2-learning</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
<b>Description</b>	Configure the timeout interval for entries in the MAC table.
<b>Default</b>	300 seconds
<b>Options</b>	<b>seconds</b> —Time elapsed before MAC table entries are timed out and entries are deleted from the table. <b>Range:</b> For MX Series routers: 10 through 1 million; for EX Series switches: 60 through 1 million
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring the MAC Table Timeout Interval on page 49</a></li><li>• <i>Configuring MAC Table Aging (CLI Procedure)</i></li></ul>

## global-no-mac-learning

---

<b>Syntax</b>	global-no-mac-learning;
<b>Hierarchy Level</b>	[edit protocols <a href="#">l2-learning</a> ], [edit protocols <a href="#">l2-learning</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
<b>Description</b>	Disable MAC learning on the entire device.
<b>Default</b>	MAC learning is enabled.
<b>Required Privilege Level</b>	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Disabling Layer 2 Learning and Forwarding on page 52</a></li><li>• <i>Understanding Q-in-Q Tunneling on EX Series Switches</i></li></ul>

## interface-mac-limit

<b>Syntax</b>	<pre>interface-mac-limit <i>limit</i> {     <b>packet-action</b> drop; }</pre>
<b>Hierarchy Level</b>	<pre>[edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit logical-systems <i>logical-system-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>switch-options</b>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> <b>switch-options</b> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> <b>switch-options</b>], [edit logical-systems <i>logical-system-name</i> <b>switch-options</b> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b>], [edit routing-instances <i>routing-instance-name</i> <b>bridge-domains</b> <i>bridge-domain-name</i> <b>bridge-options</b> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> <b>switch-options</b>], [edit routing-instances <i>routing-instance-name</i> <b>switch-options</b> interface <i>interface-name</i>], [edit <b>switch-options</b>], [edit <b>switch-options</b> interface <i>interface-name</i>], [edit switch-options interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> switch-options], [edit vlans <i>vlan-name</i> switch-options interface <i>interface-name</i>]</pre>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the <b>switch-options</b> statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the <b>virtual-switch</b> type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit switch-options], [edit switch-options interface <i>interface-name</i>], [edit vlans <i>vlan-name</i> switch-options], and [edit vlans <i>vlan-name</i> switch-options interface <i>interface-name</i>] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	<p>Configure a limit to the number of MAC addresses that can be learned from a bridge domain, VLAN, virtual switch, or set of bridge domains or VLANs.</p>



**NOTE:** For multichassis link aggregation (MC-LAG) peers in active-active mode, configuring the `interface-mac-limit` statement or changing the `interface-mac-limit` configuration when traffic is flowing can cause the MAC entries to be out of synchronization between the two MC-LAG peers, which might result in flooding. To avoid flooding, you must either halt traffic forwarding and then configure the `interface-mac-limit` statement or use the `commit at configuration` statement to commit the changes at the same time in both the peer nodes.

Alternatively, if flooding does occur, you can clear the bridge MAC table on both the routers or switches by using the `clear bridge mac-table` command. Running this command ensures that the MAC entries are re-learned and in synchronization between both the peers.

**Default** For an access port, the default MAC limit is 1024 MAC addresses. For a trunk port, the default MAC limit is 8192 MAC addresses.

**Options** *limit*—Maximum number of MAC addresses learned from an interface.

**Range:** 1 through 524287 MAC addresses per interface

The remaining statement is explained separately.

**Required Privilege Level** routing—To view this statement in the configuration.  
routing-control—To add this statement to the configuration.

**Related Documentation**

- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- [Layer 2 Learning and Forwarding for VLANs Overview](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- [Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port](#)

## packet-action

**Syntax** `packet-action action;`

**Hierarchy Level** [edit **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit logical-systems *logical-system-name* **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit protocols **l2-learning** **global-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* **bridge-domains** *bridge-domain-name* **bridge-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* protocols evpn interface-mac-limit (VPLS)],  
 [edit routing-instances *routing-instance-name* protocols evpn interface *interface-name* interface-mac-limit (VPLS)],  
 [edit routing-instances *routing-instance-name* protocols evpn mac-table-size *limit*],  
 [edit routing-instances *routing-instance-name* **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit routing-instances *routing-instance-name* **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit **switch-options** interface *interface-name* **interface-mac-limit** *limit*],  
 [edit switch-options **interface-mac-limit** *limit*],  
 [edit switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit switch-options **interface-mac-limit** *limit*],  
 [edit switch-options **mac-table-size** *limit*],  
 [edit switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options **mac-table-size** *limit*],  
 [edit vlans *vlan-name* switch-options **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options interface *interface-name* **interface-mac-limit** *limit*],  
 [edit vlans *vlan-name* switch-options **mac-table-size** *limit*]

**Release Information** Statement introduced in Junos OS Release 8.4.  
 Support for the **switch-options** statement added in Junos OS Release 9.2.  
 Support for top-level configuration for the **virtual-switch** type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.

Support for logical systems added in Junos OS Release 9.6.

[edit switch-options interface *interface-name* interface-mac-limit *limit*], [edit switch-options interface-mac-limit *limit*], [edit switch-options mac-table-size *limit*], [edit vlans *vlan-name* switch-options interface *interface-name* interface-mac-limit *limit*], [edit vlans *vlan-name* switch-options interface-mac-limit *limit*], and [edit vlans *vlan-name* switch-options mac-table-size *limit*] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.

Support for EVPNs introduced in Junos OS Release 13.2 on MX Series 3D Universal Edge Routers.

Support at the [edit switch-options interface *interface-name* interface-mac-limit *limit*] hierarchy level and hierarchy levels under [edit vlans *vlan-name*] introduced in Junos OS Release 13.2X50-D10 for EX Series switches and Junos OS Release 13.2 for the QFX Series.

**Description** Specify the action taken when packets with new source MAC addresses are received after the MAC address limit is reached. If this statement is not configured, packets with new source MAC addresses are forwarded by default.

**Default**



**NOTE:** On a QFX Series Virtual Chassis, if you include the shutdown option at the [edit vlans *vlan-name* switch-options interface *interface-name* interface-mac-limit packet-action] hierarchy level and issue the commit operation, the system generates a commit error. The system does not generate an error if you include the shutdown option at the [edit switch-options interface *interface-name* interface-mac-limit packet-action] hierarchy level.

Disabled. The default is for packets for new source MAC addresses to be forwarded after the MAC address limit is reached.

**Options**

**drop**—Drop packets with new source MAC addresses, and do not learn the new source MAC addresses.

**drop-and-log**—(EX Series switches and QFX Series only) Drop packets with new source MAC addresses, and generate an alarm, an SNMP trap, or a system log entry.

**log**—(EX Series switches and QFX Series only) Hold packets with new source MAC addresses, and generate an alarm, an SNMP trap, or a system log entry.

**none**—(EX Series switches and QFX Series only) Forward packets with new source MAC addresses, and learn the new source MAC address.

**shutdown**—(EX Series switches and QFX Series only) Disable the specified interface, and generate an alarm, an SNMP trap, or a system log entry.

**Required Privilege Level**

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.



**Related  
Documentation**

- *Configuring EVPN Routing Instances*
- *Configuring EVPN Routing Instances on EX9200 Switches*
- *Configuring MAC Limiting (CLI Procedure)*
- *Configuring Persistent MAC Learning (CLI Procedure)*
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 5](#)
- *Layer 2 Learning and Forwarding for VLANs Overview*
- [Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 5](#)
- *Layer 2 Learning and Forwarding for VLANs Overview*
- *Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port*



## PART 3

# Administration

- [Operational Mode Commands for Bridge Domains on page 117](#)
- [Operational Mode Commands for Layer 2 Learning on page 177](#)



## CHAPTER 13

# Operational Mode Commands for Bridge Domains

- clear bridge mac-table
- clear interfaces mac-database
- clear interfaces mac-database statistics
- show bridge domain
- show bridge flood
- show bridge mac-table
- show bridge statistics
- show interfaces queue

## clear bridge mac-table

---

<b>Syntax</b>	<code>clear bridge mac-table &lt;bridge-domain (all   <i>bridge-domain-name</i>)&gt; &lt;instance <i>instance-name</i>&gt; &lt;interface <i>interface-name</i>&gt; &lt;learning-vlan id (all-vlan   <i>learning-vlan-id</i>)&gt; &lt;<i>mac-address</i>&gt;</code>
<b>Release Information</b>	Command introduced in Junos OS Release 8.4.
<b>Description</b>	(MX Series routers only) Clear learned Layer 2 address information from the media access control (MAC) address table.
<b>Options</b>	<p><b>none</b>—Clear all learned Layer 2 address information from the MAC address table.</p> <p><b>bridge-domain (all   <i>bridge-domain-name</i>)</b>—(Optional) Clear learned Layer 2 MAC addresses for all bridging domains or for the specified bridging domain.</p> <p><b>instance <i>instance-name</i></b>—(Optional) Clear learned Layer 2 MAC addresses for the specified routing instance.</p> <p><b>interface <i>interface-name</i></b>—(Optional) Clear learned Layer 2 MAC addresses for the specified interface.</p> <p><b>learning-vlan-id (all-vlan   <i>learning-vlan-id</i>)</b>—(Optional) Clears learned Layer 2 MAC addresses for all VLANs or for the specified VLAN.</p> <p><b><i>mac-address</i></b>—(Optional) Clear the specified learned Layer 2 address from the MAC address table.</p>
<b>Required Privilege Level</b>	clear
<b>List of Sample Output</b>	<a href="#">clear bridge mac-table on page 118</a>
<b>Output Fields</b>	When you enter this command, you are provided feedback on the status of your request.

### Sample Output

#### clear bridge mac-table

```
user@host> clear bridge mac-table
```

## clear interfaces mac-database

---

<b>Syntax</b>	clear interfaces mac-database ( <i>interface-name</i>   <i>aex</i> ) <mac-address <i>mac-address</i> >
<b>Release Information</b>	Command introduced in Junos OS Release 8.3. Support for statement with the <b>aex</b> option introduced in Junos OS Release 15.1.
<b>Description</b>	Clear learned media access control (MAC) addresses from the hardware and MAC database for Gigabit Ethernet IQ2 interfaces or aggregated Ethernet interfaces. Static MAC addresses configured by the operator are not cleared.
<b>Options</b>	<p><b>interface-name</b>—Name of a physical or logical interface. When you clear a physical interface, all learned MAC addresses on all the logical interfaces under the physical interface are cleared.</p> <p><b>aex</b>—Name of aggregated Ethernet interface.</p> <p><b>mac-address <i>mac-address</i></b>—(Optional) Clear only the specified MAC address.</p>
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">clear interfaces mac-database on page 119</a>
<b>Output Fields</b>	This command produces no output.

## Sample Output

### clear interfaces mac-database

```
user@host> clear interfaces mac-database ge-0/0/0.0
```

## clear interfaces mac-database statistics

---

<b>Syntax</b>	<code>clear interfaces mac-database statistics (<i>interface-name</i>  all)</code>
<b>Release Information</b>	Command introduced in Junos OS Release 8.3.
<b>Description</b>	Clear statistics that are collected for every MAC address, including policer statistics, on a physical or logical interface or all interfaces.
<b>Options</b>	<code>(<i>interface-name</i>   all)</code> —Clear MAC database statistics for the specified physical or logical gigabit or 10-Gigabit Ethernet interface. Specify <b>all</b> to clear the MAC database statistics for all interfaces.
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">clear interfaces mac-database statistics (Gigabit Ethernet) on page 120</a>
<b>Output Fields</b>	This command produces no output.

### Sample Output

#### clear interfaces mac-database statistics (Gigabit Ethernet)

```
user@host> clear interfaces mac-database statistics ge-0/1/0
```



## show bridge domain

<b>Syntax</b>	<pre>show bridge domain &lt;brief   detail   extensive&gt; &lt;bridge-domain (all   domain-name)&gt; &lt;instance instance-name&gt; &lt;operational&gt;</pre>
<b>Release Information</b>	Command introduced in Junos OS Release 8.4.
<b>Description</b>	(MX Series routers only) Display bridge domain information.
<b>Options</b>	<p><b>none</b>—Display information for all bridge domains.</p> <p><b>brief   detail   extensive</b>—(Optional) Display the specified level of output.</p> <p><b>bridge-domain (all   domain-name)</b>— (Optional) Display information about all bridge domains or the specified bridge domain.</p> <p><b>instance instance-name</b>—(Optional) Display information for the specified routing instance.</p> <p><b>operational</b>—(Optional) Display information for the operational routing instances.</p>
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show bridge domain on page 121</a> <a href="#">show bridge domain brief on page 121</a> <a href="#">show bridge domain detail on page 121</a>

## Sample Output

### show bridge domain

```
user@host> show bridge domain
Instance      Bridging Domain      Type      Active
Primary Table
vs1           vlan100              bridge    2
              bridge.0
vs1           vlan200              bridge    0
              bridge.0
```

### show bridge domain brief

```
user@host> show bridge domain brief
Instance      Bridging Domain      Type      Active
Primary Table
vs1           vlan100              bridge    2
              bridge.0
vs1           vlan200              bridge    0
              bridge.0
```

### show bridge domain detail

```
user@host> show bridge domain detail
```

```

Routing Instance:vs1
  Bridging Domain:vlan100
  Router ID: 0.0.0.0
  Type: bridge                      State: Active
  Interfaces:
    ge-11/0/3.0
    ge-11/1/4.100
    ge-11/1/1.100
    ge-11/1/0.100
    xe-10/2/0.100
    xe-10/0/0.100
  Tables:
    bridge.0                        : 2 macs (2 active)
Routing Instance:vs1
  Bridging Domain:vlan200
  Router ID: 0.0.0.0
  Type: bridge                      State: Active
  Interfaces:
    ge-11/1/0.200
    ge-11/1/1.200
    ge-11/1/4.200
    xe-10/0/0.200
    xe-10/2/0.200
  Tables:
    bridge.0                        : 0 macs (0 active)

```

## show bridge flood

<b>Syntax</b>	<pre>show bridge flood &lt;brief   detail   extensive&gt; &lt;bridge-domain <i>domain-name</i>&gt; &lt;event-queue&gt; &lt;instance <i>instance-name</i>&gt; &lt;route (all-ce-flood   all ve-flood   alt-root-flood   bd-flood   mlp-flood   re-flood)&gt;</pre>
<b>Release Information</b>	Command introduced in Junos OS Release 8.4.
<b>Description</b>	(MX Series routers only) Display bridging flooding information.
<b>Options</b>	<p><b>none</b>—Display all bridging flooding information for all bridging domains.</p> <p><b>brief   detail   extensive</b>—(Optional) Display the specified level of output.</p> <p><b>bridge-domain <i>domain-name</i></b>—(Optional) Display bridging flooding information for the specified bridge domain.</p> <p><b>event-queue</b>—(Optional) Display the queue of pending bridge flood events.</p> <p><b>instance <i>instance-name</i></b>—(Optional) Display bridging flooding information for the specified routing instance.</p> <p><b>route (all-ce-flood   all ve-flood   alt-root-flood   bd-flood   mlp-flood   re-flood)</b>—(Optional) Display the following:</p> <ul style="list-style-type: none"> <li><b>all-ce-flood</b>—Display the route for flooding traffic to all customer edge routers if <b>no-local-switching</b> is enabled.</li> <li><b>all-ve-flood</b>—Display the route for flooding traffic to all VPLS edge routers if <b>no-local-switching</b> is enabled.</li> <li><b>alt-root-flood</b>—Display the Spanning Tree Protocol (STP) alt-root flooding route used for the interface.</li> <li><b>bd-flood</b>—Display the route for flooding traffic of a bridge domain if <b>no-local-switching</b> is not enabled.</li> <li><b>mlp-flood</b>—Display the route for flooding traffic to MAC learning chips.</li> <li><b>re-flood</b>—Display the route for Routing Engine flooding to all interfaces.</li> </ul>
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show bridge flood on page 124</a> <a href="#">show bridge flood brief on page 124</a> <a href="#">show bridge flood detail on page 124</a> <a href="#">show bridge flood extensive on page 125</a>
<b>Output Fields</b>	to be provided

## Sample Output

### show bridge flood

```

user@host> show bridge flood
Name: __juniper_private1__
CEs: 0
VEs: 0
Flood Routes:
  Prefix   Type      Owner                NhType  NhIndex
  0x36/16  MLP_FLOOD __vs1+vlan100__      flood   426
  0x3a/16  MLP_FLOOD __vs1+vlan200__      flood   428
Name: vs1::vlan100
CEs: 6
VEs: 0
Flood Routes:
  Prefix   Type      Owner                NhType  NhIndex
  0x35/16  ALL_FLOOD __vs1+vlan100__      flood   425
  0x35/16  RE_FLOOD  __vs1+vlan100__      flood   425
  0x3780/17 ALT_ROOT_RT ge-11/0/3.0         flood   425
  0x3b80/17 ALT_ROOT_RT ge-11/1/4.100       flood   425
  0x3c80/17 ALT_ROOT_RT ge-11/1/1.100       flood   425
  0x3d80/17 ALT_ROOT_RT ge-11/1/0.100       flood   425
  0x3e80/17 ALT_ROOT_RT xe-10/2/0.100       flood   425
  0x3f80/17 ALT_ROOT_RT xe-10/0/0.100       flood   425
Name: vs1::vlan200
CEs: 5
VEs: 0
Flood Routes:
  Prefix   Type      Owner                NhType  NhIndex
  0x39/16  ALL_FLOOD __vs1+vlan200__      flood   427
  0x39/16  RE_FLOOD  __vs1+vlan200__      flood   427
  0x4180/17 ALT_ROOT_RT ge-11/1/0.200       flood   427
  0x4080/17 ALT_ROOT_RT ge-11/1/1.200       flood   427
  0x4280/17 ALT_ROOT_RT ge-11/1/4.200       flood   427
  0x4480/17 ALT_ROOT_RT xe-10/0/0.200       flood   427
  0x4380/17 ALT_ROOT_RT xe-10/2/0.200       flood   427

```

### show bridge flood brief

```

user@host> show bridge flood brief
Name                Active CEs    Active VEs
__juniper_private1__ 0              0
vs1::vlan100         6              0
vs1::vlan200         5              0

```

### show bridge flood detail

```

user@host> show bridge flood detail
Name: __juniper_private1__
CEs: 0
VEs: 0
Flood Routes:
  Prefix   Type      Owner                NhType  NhIndex
  0x36/16  MLP_FLOOD __vs1+vlan100__      flood   426
  0x3a/16  MLP_FLOOD __vs1+vlan200__      flood   428
Name: vs1::vlan100
CEs: 6
VEs: 0
Flood Routes:
  Prefix   Type      Owner                NhType  NhIndex

```

```

0x35/16  ALL_FLOOD    __vs1+vlan100__  flood    425
0x35/16  RE_FLOOD     __vs1+vlan100__  flood    425
0x3780/17 ALT_ROOT_RT ge-11/0/3.0      flood    425
0x3b80/17 ALT_ROOT_RT ge-11/1/4.100    flood    425
0x3c80/17 ALT_ROOT_RT ge-11/1/1.100    flood    425
0x3d80/17 ALT_ROOT_RT ge-11/1/0.100    flood    425
0x3e80/17 ALT_ROOT_RT xe-10/2/0.100     flood    425
0x3f80/17 ALT_ROOT_RT xe-10/0/0.100     flood    425
Name: vs1::vlan200
CEs: 5
VEs: 0
Flood Routes:
  Prefix  Type      Owner          NhType  NhIndex
  0x39/16  ALL_FLOOD __vs1+vlan200__ flood    427
  0x39/16  RE_FLOOD  __vs1+vlan200__ flood    427
  0x4180/17 ALT_ROOT_RT ge-11/1/0.200    flood    427
  0x4080/17 ALT_ROOT_RT ge-11/1/1.200    flood    427
  0x4280/17 ALT_ROOT_RT ge-11/1/4.200    flood    427
  0x4480/17 ALT_ROOT_RT xe-10/0/0.200     flood    427
  0x4380/17 ALT_ROOT_RT xe-10/2/0.200     flood    427

```

### show bridge flood extensive

```

user@host> show bridge flood extensive
Name: __juniper_private1__
CEs: 0
VEs: 0
Flood route prefix: 0x36/16
Flood route type: MLP_FLOOD
Flood route owner: __vs1+vlan100__
Nexthop type: flood
Nexthop index: 426
  Interfaces Flooding to:
    Name          Type      NhType      Index
    1c-11/0/0.32769 LC
    1c-10/2/0.32769 LC
    1c-10/0/0.32769 LC
    1c-11/1/0.32769 LC

Flood route prefix: 0x3a/16
Flood route type: MLP_FLOOD
Flood route owner: __vs1+vlan200__
Nexthop type: flood
Nexthop index: 428
  Interfaces Flooding to:
    Name          Type      NhType      Index
    1c-10/0/0.32769 LC
    1c-10/2/0.32769 LC
    1c-11/1/0.32769 LC
Name: vs1::vlan100
CEs: 6
VEs: 0

Flood route prefix: 0x35/16
Flood route type: ALL_FLOOD
Flood route owner: __vs1+vlan100__
Nexthop type: flood
Nexthop index: 425
  Interfaces Flooding to:
    Name          Type      NhType      Index
    ge-11/0/3.0    CE

```

```

ge-11/1/4.100    CE
ge-11/1/1.100    CE
ge-11/1/0.100    CE
xe-10/2/0.100    CE
xe-10/0/0.100    CE

```

```

Flood route prefix: 0x35/16
Flood route type: RE_FLOOD
Flood route owner: __vs1+vlan100__
Nexthop type: flood
Nexthop index: 425

```

```

  Interfaces Flooding to:
  Name                Type                NhType                Index
  ge-11/0/3.0         CE
  ge-11/1/4.100       CE
  ge-11/1/1.100       CE
  ge-11/1/0.100       CE
  xe-10/2/0.100       CE
  xe-10/0/0.100       CE

```

```

Flood route prefix: 0x3780/17
Flood route type: ALT_ROOT_RT
Flood route owner: ge-11/0/3.0
Nexthop type: flood
Nexthop index: 425

```

```

  Interfaces Flooding to:
  Name                Type                NhType                Index
  ge-11/0/3.0         CE
  ge-11/1/4.100       CE
  ge-11/1/1.100       CE
  ge-11/1/0.100       CE
  xe-10/2/0.100       CE
  xe-10/0/0.100       CE

```

```

Flood route prefix: 0x3b80/17
Flood route type: ALT_ROOT_RT
Flood route owner: ge-11/1/4.100
Nexthop type: flood
Nexthop index: 425

```

```

  Interfaces Flooding to:
  Name                Type                NhType                Index
  ge-11/0/3.0         CE
  ge-11/1/4.100       CE
  ge-11/1/1.100       CE
  ge-11/1/0.100       CE
  xe-10/2/0.100       CE
  xe-10/0/0.100       CE

```

```

Flood route prefix: 0x3c80/17
Flood route type: ALT_ROOT_RT
Flood route owner: ge-11/1/1.100
Nexthop type: flood
Nexthop index: 425

```

```

  Interfaces Flooding to:
  Name                Type                NhType                Index
  ge-11/0/3.0         CE
  ge-11/1/4.100       CE
  ge-11/1/1.100       CE
  ge-11/1/0.100       CE
  xe-10/2/0.100       CE
  xe-10/0/0.100       CE

```

Flood route prefix: 0x3d80/17  
 Flood route type: ALT\_ROOT\_RT  
 Flood route owner: ge-11/1/0.100  
 Nexthop type: flood  
 Nexthop index: 425

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/0/3.0	CE		
ge-11/1/4.100	CE		
ge-11/1/1.100	CE		
ge-11/1/0.100	CE		
xe-10/2/0.100	CE		
xe-10/0/0.100	CE		

Flood route prefix: 0x3e80/17  
 Flood route type: ALT\_ROOT\_RT  
 Flood route owner: xe-10/2/0.100  
 Nexthop type: flood  
 Nexthop index: 425

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/0/3.0	CE		
ge-11/1/4.100	CE		
ge-11/1/1.100	CE		
ge-11/1/0.100	CE		
xe-10/2/0.100	CE		
xe-10/0/0.100	CE		

Flood route prefix: 0x3f80/17  
 Flood route type: ALT\_ROOT\_RT  
 Flood route owner: xe-10/0/0.100  
 Nexthop type: flood  
 Nexthop index: 425

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/0/3.0	CE		
ge-11/1/4.100	CE		
ge-11/1/1.100	CE		
ge-11/1/0.100	CE		
xe-10/2/0.100	CE		
xe-10/0/0.100	CE		

Name: vs1::vlan200

CEs: 5

VEs: 0

Flood route prefix: 0x39/16  
 Flood route type: ALL\_FLOOD  
 Flood route owner: \_\_vs1+vlan200\_\_  
 Nexthop type: flood  
 Nexthop index: 427

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/1/0.200	CE		
ge-11/1/1.200	CE		
ge-11/1/4.200	CE		
xe-10/0/0.200	CE		
xe-10/2/0.200	CE		

Flood route prefix: 0x39/16  
 Flood route type: RE\_FLOOD

Flood route owner: \_\_vs1+vlan200\_\_

Nexthop type: flood

Nexthop index: 427

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/1/0.200	CE		
ge-11/1/1.200	CE		
ge-11/1/4.200	CE		
xe-10/0/0.200	CE		
xe-10/2/0.200	CE		

Flood route prefix: 0x4180/17

Flood route type: ALT\_ROOT\_RT

Flood route owner: ge-11/1/0.200

Nexthop type: flood

Nexthop index: 427

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/1/0.200	CE		
ge-11/1/1.200	CE		
ge-11/1/4.200	CE		
xe-10/0/0.200	CE		
xe-10/2/0.200	CE		

Flood route prefix: 0x4080/17

Flood route type: ALT\_ROOT\_RT

Flood route owner: ge-11/1/1.200

Nexthop type: flood

Nexthop index: 427

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/1/0.200	CE		
ge-11/1/1.200	CE		
ge-11/1/4.200	CE		
xe-10/0/0.200	CE		
xe-10/2/0.200	CE		

Flood route prefix: 0x4280/17

Flood route type: ALT\_ROOT\_RT

Flood route owner: ge-11/1/4.200

Nexthop type: flood

Nexthop index: 427

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/1/0.200	CE		
ge-11/1/1.200	CE		
ge-11/1/4.200	CE		
xe-10/0/0.200	CE		
xe-10/2/0.200	CE		

Flood route prefix: 0x4480/17

Flood route type: ALT\_ROOT\_RT

Flood route owner: xe-10/0/0.200

Nexthop type: flood

Nexthop index: 427

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/1/0.200	CE		
ge-11/1/1.200	CE		
ge-11/1/4.200	CE		
xe-10/0/0.200	CE		



xe-10/2/0.200 CE

Flood route prefix: 0x4380/17

Flood route type: ALT\_ROOT\_RT

Flood route owner: xe-10/2/0.200

Nexthop type: flood

Nexthop index: 427

Interfaces Flooding to:

Name	Type	NhType	Index
ge-11/1/0.200	CE		
ge-11/1/1.200	CE		
ge-11/1/4.200	CE		
xe-10/0/0.200	CE		
xe-10/2/0.200	CE		

## show bridge mac-table

---

<b>Syntax</b>	<b>show bridge mac-table</b> <brief   count   detail   extensive> <bridge-domain (all   <i>bridge-domain-name</i> )> <global-count> <interface <i>interface-name</i> > <mac-address> <vlan-id (all-vlan   <i>vlan-id</i> )>
<b>Release Information</b>	Command introduced in Junos OS Release 8.4. Command introduced in Junos OS Release 15.1
<b>Description</b>	(MX Series routers only) Display Layer 2 MAC address information.
<b>Options</b>	<b>none</b> —Display all learned Layer 2 MAC address information.  <b>brief   count   detail   extensive</b> —(Optional) Display the specified level of output.  <b>bridge-domain (all   <i>bridge-domain-name</i>)</b> —(Optional) Display learned Layer 2 MAC addresses for all bridging domains or for the specified bridging domain.  <b>global-count</b> —(Optional) Display the total number of learned Layer 2 MAC addresses on the system.  <b>instance <i>instance-name</i></b> —(Optional) Display learned Layer 2 MAC addresses for the specified routing instance.  <b>interface <i>interface-name</i></b> —(Optional) Display learned Layer 2 MAC addresses for the specified interface.  <b>mac-address</b> —(Optional) Display the specified learned Layer 2 MAC address information.  <b>vlan-id (all-vlan   <i>vlan-id</i>)</b> —(Optional) Display learned Layer 2 MAC addresses for all VLANs or for the specified VLAN.
<b>Additional Information</b>	When Layer 2 protocol tunneling is enabled, the tunneling MAC address 01:00:0c:cd:cd:d0 is installed in the MAC table. When the Cisco Discovery Protocol (CDP), Spanning Tree Protocol (STP), or VLAN Trunk Protocol (VTP) is configured for Layer 2 protocol tunneling on an interface, the corresponding protocol MAC address is installed in the MAC table.
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show bridge mac-table on page 132</a> <a href="#">show bridge mac-table (with Layer 2 Services over GRE Interfaces) on page 132</a> <a href="#">show bridge mac-table (with VXLAN Enabled) on page 132</a> <a href="#">show bridge mac-table count on page 133</a> <a href="#">show bridge mac-table detail on page 133</a> <a href="#">show bridge mac-table instance pbb-evpn on page 134</a>

**Output Fields** Table 5 on page 131 describes the output fields for the **show bridge mac-table** command. Output fields are listed in the approximate order in which they appear.

**Table 5: show bridge mac-table Output fields**

Field Name	Field Description
<b>Routing instance</b>	Name of the routing instance.
<b>Bridging domain</b>	Name of the bridging domain.
<b>MAC address</b>	MAC address or addresses learned on a logical interface.
<b>MAC flags</b>	Status of MAC address learning properties for each interface: <ul style="list-style-type: none"> <li>• <b>S</b>—Static MAC address is configured.</li> <li>• <b>D</b>—Dynamic MAC address is configured.</li> <li>• <b>L</b>—Locally learned MAC address is configured.</li> <li>• <b>C</b>—Control MAC address is configured.</li> <li>• <b>SE</b>—MAC accounting is enabled.</li> <li>• <b>NM</b>—Non-configured MAC.</li> <li>• <b>R</b>—Remote PE MAC address is configured.</li> </ul>
<b>Logical interface</b>	Name of the logical interface.
<b>MAC count</b>	Number of MAC addresses learned on the specific routing instance or interface.
<b>Learning interface</b>	Name of the logical interface on which the MAC address was learned.
<b>Learning VLAN</b>	VLAN ID of the routing instance or bridge domain in which the MAC address was learned.
<b>VXLAN ID/VXLAN</b>	VXLAN Network Identifier (VNI)
<b>Layer 2 flags</b>	Debugging flags signifying that the MAC address is present in various lists.
<b>Epoch</b>	Spanning Tree Protocol epoch number identifying when the MAC address was learned. Used for debugging.
<b>Sequence number</b>	Sequence number assigned to this MAC address. Used for debugging.
<b>Learning mask</b>	Mask of the Packet Forwarding Engines where this MAC address was learned. Used for debugging.
<b>IPC generation</b>	Creation time of the logical interface when this MAC address was learned. Used for debugging.

## Sample Output

### show bridge mac-table

```
user@host> show bridge mac-table
MAC flags (S -static MAC, D -dynamic MAC, L -locally learned, C -Control MAC
          SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)

Routing instance : default-switch
Bridging domain : test1, VLAN : 1
  MAC          MAC          Logical      NH      RTR
  address      flags      interface  Index   ID
01:00:0c:cc:cc:cc S,NM      NULL
01:00:0c:cc:cc:cd S,NM      NULL
01:00:0c:cd:cd:d0 S,NM      NULL
64:87:88:6a:17:d0 D          ae0.1
64:87:88:6a:17:f0 D          ae0.1
```

### show bridge mac-table (with Layer 2 Services over GRE Interfaces)

```
user@host> show bridge mac-table
MAC flags (S -static MAC, D -dynamic MAC, L -locally learned
          SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)

Routing instance : default-switch
Bridging domain : vlan-1, VLAN : 1
  MAC          MAC          Logical      NH      RTR
  address      flags      interface  Index   ID
00:01:01:00:01:f7 D,SE      gr-1/2/10.0
00:03:00:32:01:f7 D,SE      gr-1/2/10.0
00:00:21:11:11:10 DL          ge-1/0/0.0
00:00:21:11:11:11 DL          ge-1/1/0.0

Routing instance : default-switch
Bridging domain : vlan-2, VLAN : 2
  MAC          MAC          Logical      NH      RTR
  address      flags      interface  Index   ID
00:02:01:33:01:f7 D,SE      gr-1/2/10.1
00:00:21:11:21:10 DL          ge-1/0/0.1
00:00:21:11:21:11 DL          ge-1/1/0.1
```

### show bridge mac-table (with VXLAN Enabled)

```
user@host> show bridge mac-table
MAC flags (S -static MAC, D -dynamic MAC, L -locally learned
          SE -Statistics enabled, NM -Non configured MAC, R -Remote PE MAC)

Routing instance : default-switch
Bridging domain : vlan-1, VLAN : 1
VXLAN: Id : 100, Multicast group: 226.1.1.1
  MAC          MAC          Logical      NH      RTR
  address      flags      interface  Index   ID
00:01:01:00:01:f7 D,SE      vtep.1052010
00:03:00:32:01:f7 D,SE      vtep.1052011
00:00:21:11:11:10 DL          ge-1/0/0.0
00:00:21:11:11:11 DL          ge-1/1/0.0
```

```

Routing instance : default-switch
Bridging domain : vlan-2, VLAN : 2, VXLAN : 200
VXLAN: Id : 200, Multicast group: 226.1.1.2
MAC          MAC          Logical
address      flags      interface
00:02:01:33:01:f7 D,SE    vtep.1052010
00:04:00:14:01:f7 D,SE    vtep.1052011
00:00:21:11:21:10 DL      ge-1/0/0.1
00:00:21:11:21:11 DL      ge-1/1/0.1

```

### show bridge mac-table count

```

user@host> show bridge mac-table count
2 MAC address learned in routing instance vs1 bridge domain vlan100

MAC address count per interface within routing instance:
Logical interface      MAC count
ge-11/0/3.0            1
ge-11/1/4.100          0
ge-11/1/1.100          0
ge-11/1/0.100          0
xe-10/2/0.100          1
xe-10/0/0.100          0

MAC address count per learn VLAN within routing instance:
Learn VLAN ID          MAC count
0                        2

0 MAC address learned in routing instance vs1 bridge domain vlan200

MAC address count per interface within routing instance:
Logical interface      MAC count
ge-11/1/0.200          0
ge-11/1/1.200          0
ge-11/1/4.200          0
xe-10/0/0.200          0
xe-10/2/0.200          0

MAC address count per learn VLAN within routing instance:
Learn VLAN ID          MAC count
0                        0

```

### show bridge mac-table detail

```

user@host> show bridge mac-table detail
MAC address: 00:00:00:19:1c:db
Routing instance: vs1
Bridging domain: vlan100
Learning interface: ge-11/0/3.0   Learning VLAN: 0
Layer 2 flags: in_ifd, in_ifl, in_vlan, kernel
Epoch: 4                          Sequence number: 0
Learning mask: 0x800               IPC generation: 0

MAC address: 00:00:00:59:3a:2f
Routing instance: vs1
Bridging domain: vlan100
Learning interface: xe-10/2/0.100 Learning VLAN: 0
Layer 2 flags: in_ifd, in_ifl, in_vlan, kernel
Epoch: 7                          Sequence number: 0
Learning mask: 0x400               IPC generation: 0

```

**show bridge mac-table instance pbb-evpn**

```
user@host> show bridge mac-table instance pbb-evpn
Routing instance : pbb-evpn
Bridging domain : isid-bd10000, ISID : 10000
MAC          MAC      Logical      NH      RTR
address      flags   interface   Index   ID
00:19:e2:b0:76:eb  D      cbp.1000
aa:bb:cc:dd:ee:f2  DC
aa:bb:cc:dd:ee:f3  DC          1048576 1048576
          1048575 1048575
```

## show bridge statistics

<b>Syntax</b>	<b>show bridge statistics</b> <b>&lt;bridge-domain <i>domain-name</i>&gt;</b> <b>&lt;instance <i>instance-name</i>&gt;</b>
<b>Release Information</b>	Command introduced in Junos OS Release 8.4.
<b>Description</b>	(MX Series routers only) Display bridge statistics.
<b>Options</b>	<b>none</b> —Display bridge statistics for all bridge domains in all routing instances.  <b>bridge-domain <i>domain-name</i></b> —(Optional) Display statistics for the specified bridge domain.  <b>instance <i>instance-name</i></b> —(Optional) Display statistics for the specified routing instance.
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show bridge statistics on page 135</a>

## Sample Output

### show bridge statistics

```

user@host> show bridge statistics
Information for routing instance:

Routing instance : __juniper_private1__
  Index: 1                      Sequence number: 0
  MAC limit: 5000                MACs learned: 0
  Static MACs learned: 0         Non config Static MACs learned: 0
  Handle: 0x829e800

Information for routing instance:

Routing instance : vs1
  Bridging domain : vlan100
  Index: 3                      Sequence number: 0
  MAC limit: 5120                MACs learned: 2
  Static MACs learned: 0         Non config Static MACs learned: 0
  Handle: 0x829e400
  Flags: Bridge instance, Config defined, VLAN : 100
  Local interface: ge-11/0/3.0, Index: 79
    Broadcast packets: 1
    Broadcast bytes : 65
    Multicast packets: 0
    Multicast bytes : 0
    Flooded packets : 0
    Flooded bytes : 0
    Unicast packets : 358624489
    Unicast bytes : 23310592305
    Current MAC count: 1 (Limit 1024)
  Local interface: ge-11/1/4.100, Index: 84
    Broadcast packets: 0
    Broadcast bytes : 0
    Multicast packets: 0
    Multicast bytes : 0

```

```

Flooded packets : 0
Flooded bytes : 0
Unicast packets : 0
Unicast bytes : 0
Current MAC count: 0 (Limit 1024)
Local interface: ge-11/1/1.100, Index: 86
Broadcast packets: 0
Broadcast bytes : 0
Multicast packets: 0
Multicast bytes : 0
Flooded packets : 0
Flooded bytes : 0
Unicast packets : 0
Unicast bytes : 0
Current MAC count: 0 (Limit 1024)
Local interface: ge-11/1/0.100, Index: 87
Broadcast packets: 0
Broadcast bytes : 0
Multicast packets: 0
Multicast bytes : 0
Flooded packets : 0
Flooded bytes : 0
Unicast packets : 0
Unicast bytes : 0
Current MAC count: 0 (Limit 1024)
Local interface: xe-10/2/0.100, Index: 88
Broadcast packets: 0
Broadcast bytes : 0
Multicast packets: 0
Multicast bytes : 0
Flooded packets : 0
Flooded bytes : 0
Unicast packets : 358627393
Unicast bytes : 23310781065
Current MAC count: 1 (Limit 1024)
Local interface: xe-10/0/0.100, Index: 89
Broadcast packets: 0
Broadcast bytes : 0
Multicast packets: 0
Multicast bytes : 0
Flooded packets : 0
Flooded bytes : 0
Unicast packets : 0
Unicast bytes : 0
Current MAC count: 0 (Limit 1024)

```

#### Information for routing instance:

```

Routing instance : vs1
Bridging domain : vlan200
Index: 4                      Sequence number: 0
MAC limit: 5120              MACs learned: 0
Static MACs learned: 0      Non config Static MACs learned: 0
Handle: 0x829e600
Flags: Bridge instance, Config defined, VLAN : 200
Local interface: ge-11/1/0.200, Index: 90
Broadcast packets: 0
Broadcast bytes : 0
Multicast packets: 0
Multicast bytes : 0
Flooded packets : 0

```



```
Flooded bytes      : 0
Unicast packets   : 0
Unicast bytes     : 0
Current MAC count: 0 (Limit 1024)
Local interface: ge-11/1/1.200, Index: 91
Broadcast packets: 0
Broadcast bytes   : 0
Multicast packets: 0
Multicast bytes   : 0
Flooded packets   : 0
Flooded bytes     : 0
Unicast packets   : 0
Unicast bytes     : 0
Current MAC count: 0 (Limit 1024)
Local interface: ge-11/1/4.200, Index: 92
Broadcast packets: 0
Broadcast bytes   : 0
Multicast packets: 0
Multicast bytes   : 0
Flooded packets   : 0
Flooded bytes     : 0
Unicast packets   : 0
Unicast bytes     : 0
Current MAC count: 0 (Limit 1024)
Local interface: xe-10/0/0.200, Index: 93
Broadcast packets: 0
Broadcast bytes   : 0
Multicast packets: 0
Multicast bytes   : 0
Flooded packets   : 0
Flooded bytes     : 0
Unicast packets   : 0
Unicast bytes     : 0
Current MAC count: 0 (Limit 1024)
Local interface: xe-10/2/0.200, Index: 94
Broadcast packets: 4
Broadcast bytes   : 260
Multicast packets: 0
Multicast bytes   : 0
Flooded packets   : 0
Flooded bytes     : 0
Unicast packets   : 0
Unicast bytes     : 0
Current MAC count: 0 (Limit 1024)
```

## show interfaces queue

---

**Syntax**    show interfaces queue  
              <aggregate | remaining-traffic>  
              <both-ingress-egress>  
              <egress>  
              <forwarding-class *forwarding-class*>  
              <ingress>  
              <interface-name *interface-name*>  
              <l2-statistics>

**Release Information**    Command introduced before Junos OS Release 7.4.  
                              **both-ingress-egress**, **egress**, and **ingress** options introduced in Junos OS Release 7.6.  
                              Command introduced in Junos OS Release 11.1 for the QFX Series.  
                              **l2-statistics** option introduced in Junos OS Release 12.1.  
                              Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

**Description**    Display class-of-service (CoS) queue information for physical interfaces.

**Options**    **none**—Show detailed CoS queue statistics for all physical interfaces.

**aggregate**—(Optional) Display the aggregated queuing statistics of all logical interfaces that have traffic-control profiles configured. (Not on the QFX Series.)

**both-ingress-egress**—(Optional) On Gigabit Ethernet Intelligent Queuing 2 (IQ2) PICs, display both ingress and egress queue statistics. (Not on the QFX Series.)

**egress**—(Optional) Display egress queue statistics.

**forwarding-class *forwarding-class***—(Optional) Forwarding class name for this queue. Shows detailed CoS statistics for the queue associated with the specified forwarding class.

**ingress**—(Optional) On Gigabit Ethernet IQ2 PICs, display ingress queue statistics. (Not on the QFX Series.)

**interface-name *interface-name***—(Optional) Show detailed CoS queue statistics for the specified interface.

**l2-statistics**—(Optional) Display Layer 2 statistics for MLPPP, FRF.15, and FRF.16 bundles

**remaining-traffic**—(Optional) Display the remaining-traffic queue statistics of all logical interfaces that have traffic-control profiles configured.

### Overhead for Layer 2 Statistics

Transmitted packets and transmitted byte counts are displayed for the Layer 2 level with the addition of encapsulation overheads applied for fragmentation, as shown in [Table 6 on page 139](#). Others counters, such as packets and bytes queued (input) and drop counters, are displayed at the Layer 3 level. In the case of link fragmentation and interleaving (LFI) for which fragmentation is not applied, corresponding Layer 2 overheads are added, as shown in [Table 6 on page 139](#).

Table 6: Layer 2 Overhead and Transmitted Packets or Byte Counts

Protocol	Fragmentation		LFI
	First fragmentation	Second to <i>n</i> fragmentations	
	Bytes	Bytes	
MLPPP (Long)	13	12	8
MLPPP (short)	11	10	8
MLFR (FRF15)	12	10	8
MFR (FRF16)	10	8	-
MCMLPPP(Long)	13	12	-
MCMLPPP(Short)	11	10	-

#### Layer 2 Statistics—Fragmentation Overhead Calculation

##### MLPPP/MC-MLPPP Overhead details:

=====

##### Fragment 1:

```

Outer PPP header           : 4 bytes
Long or short sequence MLPPP header : 4 bytes or 2 bytes
Inner PPP header           : 1 byte
HDLC flag and FCS bytes    : 4 bytes

```

##### Fragments 2 .. n :

```

Outer PPP header           : 4 bytes
Long or short sequence MLPPP header : 4 bytes or 2 bytes
HDLC flag and FCS bytes    : 4 bytes

```

##### MLFR (FRF15) Overhead details:

=====

##### Fragment 1:

```

Framereley header         : 2 bytes
Control,NLPID              : 2 bytes
Fragmentaion header       : 2 bytes
Inner proto                : 2 bytes
HDLC flag and FCS         : 4 bytes

```

##### Fragments 2 ...n :

```

Framereley header         : 2 bytes
Control,NLPID              : 2 bytes
Fragmentaion header       : 2 bytes
HDLC flag and FCS         : 4 bytes

```

##### MFR (FRF16) Overhead details:

=====

```

Fragment 1:
  Fragmentation header : 2 bytes
  Framereelay header   : 2 bytes
  Inner proto          : 2 bytes
  HDLC flag and FCS    : 4 bytes

Fragments 2 ...n :
  Fragmentation header : 2 bytes
  Framereelay header   : 2 bytes
  HDLC flag and FCS    : 4 bytes

```

## Overhead with LFI

```

MLPPP(Long & short sequence):
=====
  Outer PPP header : 4 bytes
  HDLC flag and FCS : 4 bytes

MLFR (FRF15):
=====
  Framereelay header : 2 bytes
  Control,NLPID      : 2 bytes
  HDLC flag and FCS  : 4 bytes

```

The following examples show overhead for different cases:

- A 1000-byte packet is sent to a mlppp bundle without any fragmentation. At the Layer 2 level, bytes transmitted is 1013 in 1 packet. This overhead is for MLPPP long sequence encap.
- A 1000-byte packet is sent to a mlppp bundle with a fragment threshold of 250byte. At the Layer 2 level, bytes transmitted is 1061 bytes in 5 packets.
- A 1000-byte LFI packet is sent to an mlppp bundle. At the Layer 2 level, bytes transmitted is 1008 in 1 packet.

**remaining-traffic**—(Optional) Display the queuing statistics of all logical interfaces that do not have traffic-control profiles configured. (Not on the QFX Series.)

## Additional Information

For rate-limited interfaces hosted on Modular Interface Cards (MICs), Modular Port Concentrators (MPCs), or Enhanced Queuing DPCs, rate-limit packet-drop operations occur *before* packets are queued for transmission scheduling. For such interfaces, the statistics for queued traffic do not include the packets that have already been dropped due to rate limiting, and consequently the displayed statistics for queued traffic are the same as the displayed statistics for transmitted traffic.



**NOTE:** For rate-limited interfaces hosted on other types of hardware, rate-limit packet-drop operations occur *after* packets are queued for transmission scheduling. For these other interface types, the statistics for queued traffic include the packets that are later dropped due to rate limiting, and consequently the displayed statistics for queued traffic equals the sum of the statistics for transmitted and rate-limited traffic.

On M Series routers (except for the M320 and M120 routers), this command is valid only for a PIC installed on an enhanced Flexible PIC Concentrator (FPC).

Queue statistics for aggregated interfaces are supported on the M Series and T Series routers only. Statistics for an aggregated interface are the summation of the queue statistics of the child links of that aggregated interface. You can view the statistics for a child interface by using the **show interfaces statistics** command for that child interface.

When you configure tricolor marking on a 10-port 1-Gigabit Ethernet PIC, for queues 6 and 7 only, the output does not display the number of queued bytes and packets, or the number of bytes and packets dropped because of RED. If you do not configure tricolor marking on the interface, these statistics are available for all queues.

For the 4-port Channelized OC12 IQE PIC and 1-port Channelized OC48 IQE PIC, the **Packet Forwarding Engine Chassis Queues** field represents traffic bound for a particular physical interface on the PIC. For all other PICs, the **Packet Forwarding Engine Chassis Queues** field represents the total traffic bound for the PIC.

For Gigabit Ethernet IQ2 PICs, the **show interfaces queue** command output does not display the number of tail-dropped packets. This limitation does not apply to Packet Forwarding Engine chassis queues.

When fragmentation occurs on the egress interface, the first set of packet counters shows the postfragmentation values. The second set of packet counters (under the **Packet Forwarding Engine Chassis Queues** field) shows the prefragmentation values.

The behavior of the **egress** queues for the **Routing Engine-Generated Traffic** is not same as the configured queue for MLPPP and MFR configurations.

For information about how to configure CoS, see the *Junos OS Network Interfaces Library for Routing Devices*. For related CoS operational mode commands, see the [CLI Explorer](#).

<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show interfaces queue (Rate-Limited Interface on a Gigabit Ethernet MIC in an MPC) on page 146</a> <a href="#">show interfaces queue (Aggregated Ethernet on a T320 Router) on page 147</a> <a href="#">show interfaces queue (Gigabit Ethernet on a T640 Router) on page 149</a> <a href="#">show interfaces queue aggregate (Gigabit Ethernet Enhanced DPC) on page 149</a> <a href="#">show interfaces queue (Gigabit Ethernet IQ2 PIC) on page 153</a> <a href="#">show interfaces queue both-ingress-egress (Gigabit Ethernet IQ2 PIC) on page 156</a> <a href="#">show interfaces queue ingress (Gigabit Ethernet IQ2 PIC) on page 158</a> <a href="#">show interfaces queue egress (Gigabit Ethernet IQ2 PIC) on page 159</a> <a href="#">show interfaces queue remaining-traffic (Gigabit Ethernet Enhanced DPC) on page 161</a> <a href="#">show interfaces queue (Channelized OC12 IQE Type 3 PIC in SONET Mode) on page 163</a> <a href="#">show interfaces queue (QFX Series) on page 173</a> <a href="#">show interfaces queue l2-statistics (lsq interface) on page 174</a> <a href="#">show interfaces queue lsq (lsq-ifd) on page 175</a>
<b>Output Fields</b>	Table 7 on page 142 lists the output fields for the <b>show interfaces queue</b> command. Output fields are listed in the approximate order in which they appear.

Table 7: show interfaces queue Output Fields

Field Name	Field Description
<b>Physical interface</b>	Name of the physical interface.
<b>Enabled</b>	State of the interface. Possible values are described in the "Enabled Field" section under <i>Common Output Fields Description</i> .
<b>Interface index</b>	Physical interface's index number, which reflects its initialization sequence.
<b>SNMP ifIndex</b>	SNMP index number for the interface.
<b>Forwarding classes supported</b>	Total number of forwarding classes supported on the specified interface.
<b>Forwarding classes in use</b>	Total number of forwarding classes in use on the specified interface.
<b>Ingress queues supported</b>	On Gigabit Ethernet IQ2 PICs only, total number of ingress queues supported on the specified interface.
<b>Ingress queues in use</b>	On Gigabit Ethernet IQ2 PICs only, total number of ingress queues in use on the specified interface.
<b>Output queues supported</b>	Total number of output queues supported on the specified interface.
<b>Output queues in use</b>	Total number of output queues in use on the specified interface.
<b>Egress queues supported</b>	Total number of egress queues supported on the specified interface.
<b>Egress queues in use</b>	Total number of egress queues in use on the specified interface.
<b>Queue counters (Ingress)</b>	CoS queue number and its associated user-configured forwarding class name. Displayed on IQ2 interfaces. <ul style="list-style-type: none"> <li>• <b>Queued packets</b>—Number of queued packets.</li> <li>• <b>Transmitted packets</b>—Number of transmitted packets.</li> <li>• <b>Dropped packets</b>—Number of packets dropped by the ASIC's RED mechanism.</li> </ul>
<b>Burst size</b>	(Logical interfaces on IQ PICs only) Maximum number of bytes up to which the logical interface can burst. The burst size is based on the shaping rate applied to the interface.
The following output fields are applicable to both interface component and Packet Forwarding component in the <b>show interfaces queue</b> command:	
<b>Queue</b>	Queue number.
<b>Forwarding classes</b>	Forwarding class name.

Table 7: show interfaces queue Output Fields (*continued*)

Field Name	Field Description
<b>Queued Packets</b>	<p>Number of packets queued to this queue.</p> <p><b>NOTE:</b> For Gigabit Ethernet IQ2 interfaces, the Queued Packets count is calculated by the Junos OS interpreting one frame buffer as one packet. If the queued packets are very large or very small, the calculation might not be completely accurate for transit traffic. The count is completely accurate for traffic terminated on the router.</p> <p>For rate-limited interfaces hosted on MICs or MPCs only, this statistic does not include traffic dropped due to rate limiting. For more information, see <a href="#">“Additional Information” on page 140</a>.</p>
<b>Queued Bytes</b>	<p>Number of bytes queued to this queue. The byte counts vary by interface hardware. For more information, see <a href="#">Table 8 on page 145</a>.</p> <p>For rate-limited interfaces hosted on MICs or MPCs only, this statistic does not include traffic dropped due to rate limiting. For more information, see <a href="#">“Additional Information” on page 140</a>.</p>
<b>Transmitted Packets</b>	<p>Number of packets transmitted by this queue. When fragmentation occurs on the egress interface, the first set of packet counters shows the postfragmentation values. The second set of packet counters (displayed under the <b>Packet Forwarding Engine Chassis Queues</b> field) shows the prefragmentation values.</p> <p><b>NOTE:</b> For Layer 2 statistics, see <a href="#">“Overhead for Layer 2 Statistics” on page 138</a></p>
<b>Transmitted Bytes</b>	<p>Number of bytes transmitted by this queue. The byte counts vary by interface hardware. For more information, see <a href="#">Table 8 on page 145</a>.</p> <p><b>NOTE:</b> On MX Series routers, this number can be inaccurate when you issue the command for a physical interface repeatedly and in quick succession, because the statistics for the child nodes are collected infrequently. Wait ten seconds between successive iterations to avoid this situation.</p> <p><b>NOTE:</b> For Layer 2 statistics, see <a href="#">“Overhead for Layer 2 Statistics” on page 138</a></p>
<b>Tail-dropped packets</b>	Number of packets dropped because of tail drop.
<b>RL-dropped packets</b>	<p>Number of packets dropped due to rate limiting.</p> <p>For rate-limited interfaces hosted on MICs, MPCs, and Enhanced Queuing DPCs only, this statistic is not included in the queued traffic statistics. For more information, see <a href="#">“Additional Information” on page 140</a>.</p> <p><b>NOTE:</b> The <b>RL-dropped packets</b> counter is not supported on the PTX Series Packet Transport Routers, and is omitted from the output.</p>
<b>RL-dropped bytes</b>	<p>Number of bytes dropped due to rate limiting.</p> <p>For rate-limited interfaces hosted on MICs, MPCs, and Enhanced Queuing DPCs only, this statistic is not included in the queued traffic statistics. For more information, see <a href="#">“Additional Information” on page 140</a>.</p>

Table 7: show interfaces queue Output Fields (*continued*)

Field Name	Field Description
RED-dropped packets	<p>Number of packets dropped because of random early detection (RED).</p> <ul style="list-style-type: none"> <li>(M Series and T Series routers only) On M320 and M120 routers and the T Series routers, the total number of dropped packets is displayed. On all other M Series routers, the output classifies dropped packets into the following categories: <ul style="list-style-type: none"> <li><b>Low, non-TCP</b>—Number of low-loss priority non-TCP packets dropped because of RED.</li> <li><b>Low, TCP</b>—Number of low-loss priority TCP packets dropped because of RED.</li> <li><b>High, non-TCP</b>—Number of high-loss priority non-TCP packets dropped because of RED.</li> <li><b>High, TCP</b>—Number of high-loss priority TCP packets dropped because of RED.</li> </ul> </li> <li>(MX Series routers with enhanced DPCs, and T Series routers with enhanced FPCs only) The output classifies dropped packets into the following categories: <ul style="list-style-type: none"> <li><b>Low</b>—Number of low-loss priority packets dropped because of RED.</li> <li><b>Medium-low</b>—Number of medium-low loss priority packets dropped because of RED.</li> <li><b>Medium-high</b>—Number of medium-high loss priority packets dropped because of RED.</li> <li><b>High</b>—Number of high-loss priority packets dropped because of RED.</li> </ul> </li> </ul> <p><b>NOTE:</b> Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.</p>
RED-dropped bytes	<p>Number of bytes dropped because of RED. The byte counts vary by interface hardware. For more information, see <a href="#">Table 8 on page 145</a>.</p> <ul style="list-style-type: none"> <li>(M Series and T Series routers only) On M320 and M120 routers and the T Series routers, only the total number of dropped bytes is displayed. On all other M Series routers, the output classifies dropped bytes into the following categories: <ul style="list-style-type: none"> <li><b>Low, non-TCP</b>—Number of low-loss priority non-TCP bytes dropped because of RED.</li> <li><b>Low, TCP</b>—Number of low-loss priority TCP bytes dropped because of RED.</li> <li><b>High, non-TCP</b>—Number of high-loss priority non-TCP bytes dropped because of RED.</li> <li><b>High, TCP</b>—Number of high-loss priority TCP bytes dropped because of RED.</li> </ul> </li> </ul> <p><b>NOTE:</b> Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.</p>

Byte counts vary by interface hardware. [Table 8 on page 145](#) shows how the byte counts on the outbound interfaces vary depending on the interface hardware. [Table 8 on page 145](#) is based on the assumption that outbound interfaces are sending IP traffic with 478 bytes per packet.



Table 8: Byte Count by Interface Hardware

Interface Hardware	Output Level	Byte Count Includes	Comments
Gigabit Ethernet IQ and IQE PICs	Interface	<p>Queued: 490 bytes per packet, representing 478 bytes of Layer 3 packet + 12 bytes</p> <p>Transmitted: 490 bytes per packet, representing 478 bytes of Layer 3 packet + 12 bytes</p> <p>RED dropped: 496 bytes per packet representing 478 bytes of Layer 3 packet + 18 bytes</p>	<p>The 12 additional bytes include 6 bytes for the destination MAC address + 4 bytes for the VLAN + 2 bytes for the Ethernet type.</p> <p>For RED dropped, 6 bytes are added for the source MAC address.</p>
	Packet forwarding component	<p>Queued: 478 bytes per packet, representing 478 bytes of Layer 3 packet</p> <p>Transmitted: 478 bytes per packet, representing 478 bytes of Layer 3 packet</p>	—
Non-IQ PIC	Interface	<p>T Series, TX Series, T1600, and MX Series routers:</p> <ul style="list-style-type: none"> <li>Queued: 478 bytes of Layer 3 packet.</li> <li>Transmitted: 478 bytes of Layer 3 packet.</li> </ul> <p>T4000 routers with Type 5 FPCs :</p> <ul style="list-style-type: none"> <li>Queued: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes CRC + the full Layer 1 overhead 8 bytes preamble + 12 bytes Inter frame Gap.</li> <li>Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes CRC + the full Layer 1 overhead 8 bytes preamble + 12 bytes Interframe Gap.</li> </ul> <p>M Series routers:</p> <ul style="list-style-type: none"> <li>Queued: 478 bytes of Layer 3 packet.</li> <li>Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead.</li> </ul> <p>PTX Series Packet Transport Routers:</p> <ul style="list-style-type: none"> <li>Queued: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes FCS + the full Layer 1 overhead of the MAC header DA + SA + EtherType (non-VLAN).</li> <li>Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes CRC + the full Layer 1 overhead of the MAC header DA + SA + EtherType (non-VLAN).</li> <li>RED dropped: 478 bytes of Layer 3 packet + 22 bytes special header. To the TQ, this packet has 4 bytes more than queued or transmitted.</li> </ul>	<p>The Layer 2 overhead is 14 bytes for non-VLAN traffic and 18 bytes for VLAN traffic.</p>

Table 8: Byte Count by Interface Hardware (*continued*)

Interface Hardware	Output Level	Byte Count Includes	Comments
IQ and IQE PICs with a SONET/SDH interface	Interface	<p>Queued: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes</p> <p>Transmitted: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes</p> <p>RED dropped: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes</p>	The additional 4 bytes are for the Layer 2 Point-to-Point Protocol (PPP) header.
	Packet forwarding component	<p>Queued: 478 bytes per packet, representing 478 bytes of Layer 3 packet</p> <p>Transmitted: 486 bytes per packet, representing 478 bytes of Layer 3 packet + 8 bytes</p>	For transmitted packets, the additional 8 bytes includes 4 bytes for the PPP header and 4 bytes for a cookie.
Non-IQ PIC with a SONET/SDH interface	Interface	<p>T Series, TX Series, T1600, and MX Series routers:</p> <ul style="list-style-type: none"> <li>Queued: 478 bytes of Layer 3 packet.</li> <li>Transmitted: 478 bytes of Layer 3 packet.</li> </ul> <p>M Series routers:</p> <ul style="list-style-type: none"> <li>Queued: 478 bytes of Layer 3 packet.</li> <li>Transmitted: 483 bytes per packet, representing 478 bytes of Layer 3 packet + 5 bytes</li> <li>RED dropped: 478 bytes per packet, representing 478 bytes of Layer 3 packet</li> </ul>	For transmitted packets, the additional 5 bytes includes 4 bytes for the PPP header and 1 byte for the packet loss priority (PLP).
Interfaces configured with Frame Relay Encapsulation	Interface	The default Frame Relay overhead is 7 bytes. If you configure the Frame Check Sequence (FCS) to 4 bytes, then the overhead increases to 10 bytes.	
1-port 10-Gigabit Ethernet IQ2 and IQ2-E PICs	Interface	<p>Queued: 478 bytes of Layer 3 packet + the full Layer 2 overhead including CRC.</p> <p>Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead including CRC.</p>	The Layer 2 overhead is 18 bytes for non-VLAN traffic and 22 bytes for VLAN traffic.
4-port 1G IQ2 and IQ2-E PICs	Packet forwarding component	Queued: 478 bytes of Layer 3 packet.	—
8-port 1G IQ2 and IQ2-E PICs		Transmitted: 478 bytes of Layer 3 packet.	

## Sample Output

### show interfaces queue (Rate-Limited Interface on a Gigabit Ethernet MIC in an MPC)

The following example shows queue information for the rate-limited interface ge-4/2/0 on a Gigabit Ethernet MIC in an MPC. For rate-limited queues for interfaces hosted on MICs or MPCs, rate-limit packet drops occur prior to packet output queuing. In the

command output, the nonzero statistics displayed in the **RL-dropped packets** and **RL-dropped bytes** fields quantify the traffic dropped to rate-limit queue 0 output to 10 percent of 1 gigabyte (100 megabits) per second. Because the RL-dropped traffic is not included in the **Queued** statistics, the statistics displayed for queued traffic are the same as the statistics for transmitted traffic.

```
user@host> show interfaces queue ge-4/2/0
Physical interface: ge-4/2/0, Enabled, Physical link is Up
  Interface index: 203, SNMP ifIndex: 1054
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets          :          131300649          141751 pps
    Bytes            :          11287964840        99793248 bps
  Transmitted:
    Packets          :          131300649          141751 pps
    Bytes            :          11287964840        99793248 bps
    Tail-dropped packets :              0              0 pps
    RL-dropped packets  :          205050862        602295 pps
    RL-dropped bytes    :          13595326612      327648832 bps
    RED-dropped packets :              0              0 pps
      Low              :              0              0 pps
      Medium-low       :              0              0 pps
      Medium-high      :              0              0 pps
      High             :              0              0 pps
    RED-dropped bytes   :              0              0 bps
      Low              :              0              0 bps
      Medium-low       :              0              0 bps
      Medium-high      :              0              0 bps
      High             :              0              0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets          :              0              0 pps
    Bytes            :              0              0 bps
```

#### show interfaces queue (Aggregated Ethernet on a T320 Router)

The following example shows that the aggregated Ethernet interface, **ae1**, has traffic on queues **af1** and **af12**:

```
user@host> show interfaces queue ae1
Physical interface: ae1, Enabled, Physical link is Up
  Interface index: 158, SNMP ifIndex: 33 Forwarding classes: 8 supported, 8 in use
Output queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: be
  Queued:
    Packets          :              5              0 pps
    Bytes            :              242              0 bps
  Transmitted:
    Packets          :              5              0 pps
    Bytes            :              242              0 bps
    Tail-dropped packets :              0              0 pps
    RED-dropped packets :              0              0 pps
    RED-dropped bytes   :              0              0 bps
Queue: 1, Forwarding classes: af1
  Queued:
    Packets          :          42603765          595484 pps
```

```

Bytes                :          5453281920          609776496 bps
Transmitted:
Packets              :          42603765           595484 pps
Bytes                :          5453281920          609776496 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets  :              0              0 pps
RED-dropped bytes    :              0              0 bps
Queue: 2, Forwarding classes: ef1
Queued:
Packets              :              0              0 pps
Bytes                :              0              0 bps
Transmitted:
Packets              :              0              0 pps
Bytes                :              0              0 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets  :              0              0 pps
RED-dropped bytes    :              0              0 bps
Queue: 3, Forwarding classes: nc
Queued:
Packets              :              45              0 pps
Bytes                :             3930              0 bps
Transmitted:
Packets              :              45              0 pps
Bytes                :             3930              0 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets  :              0              0 pps
RED-dropped bytes    :              0              0 bps
Queue: 4, Forwarding classes: af11
Queued:
Packets              :              0              0 pps
Bytes                :              0              0 bps
Transmitted:
Packets              :              0              0 pps
Bytes                :              0              0 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets  :              0              0 pps
RED-dropped bytes    :              0              0 bps
Queue: 5, Forwarding classes: ef11
Queued:
Packets              :              0              0 pps
Bytes                :              0              0 bps
Transmitted:
Packets              :              0              0 pps
Bytes                :              0              0 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets  :              0              0 pps
RED-dropped bytes    :              0              0 bps
Queue: 6, Forwarding classes: af12
Queued:
Packets              :          31296413           437436 pps
Bytes                :          4005940864          447935200 bps
Transmitted:
Packets              :          31296413           437436 pps
Bytes                :          4005940864          447935200 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets  :              0              0 pps
RED-dropped bytes    :              0              0 bps
Queue: 7, Forwarding classes: nc2
Queued:
Packets              :              0              0 pps
Bytes                :              0              0 bps

```

```

Transmitted:
Packets      :                0                0 pps
Bytes        :                0                0 bps
Tail-dropped packets :                0                0 pps
RED-dropped packets :                0                0 pps
RED-dropped bytes  :                0                0 bps

```

#### show interfaces queue (Gigabit Ethernet on a T640 Router)

```

user@host> show interfaces queue
Physical interface: ge-7/0/1, Enabled, Physical link is Up
  Interface index: 150, SNMP ifIndex: 42
  Forwarding classes: 8 supported, 8 in use
  Output queues: 8 supported, 8 in use
  Queue: 0, Forwarding classes: be
    Queued:
      Packets      :                13                0 pps
      Bytes        :               622                0 bps
    Transmitted:
      Packets      :                13                0 pps
      Bytes        :               622                0 bps
      Tail-dropped packets :                0                0 pps
      RED-dropped packets :                0                0 pps
      RED-dropped bytes  :                0                0 bps
  Queue: 1, Forwarding classes: af1
    Queued:
      Packets      :            1725947945            372178 pps
      Bytes        :            220921336960          381110432 bps
    Transmitted:
      Packets      :            1725947945            372178 pps
      Bytes        :            220921336960          381110432 bps
      Tail-dropped packets :                0                0 pps
      RED-dropped packets :                0                0 pps
      RED-dropped bytes  :                0                0 bps
  Queue: 2, Forwarding classes: ef1
    Queued:
      Packets      :                0                0 pps
      Bytes        :                0                0 bps
    Transmitted:
      Packets      :                0                0 pps
      Bytes        :                0                0 bps
      Tail-dropped packets :                0                0 pps
      RED-dropped packets :                0                0 pps
      RED-dropped bytes  :                0                0 bps
  Queue: 3, Forwarding classes: nc
    Queued:
      Packets      :                571                0 pps
      Bytes        :               49318               336 bps
    Transmitted:
      Packets      :                571                0 pps
      Bytes        :               49318               336 bps
      Tail-dropped packets :                0                0 pps
      RED-dropped packets :                0                0 pps
      RED-dropped bytes  :                0                0 bps

```

#### show interfaces queue aggregate (Gigabit Ethernet Enhanced DPC)

```

user@host> show interfaces queue ge-2/2/9 aggregate

```

```

Physical interface: ge-2/2/9, Enabled, Physical link is Up
  Interface index: 238, SNMP ifIndex: 71
  Forwarding classes: 16 supported, 4 in use
  Ingress queues: 4 supported, 4 in use
  Queue: 0, Forwarding classes: best-effort
    Queued:
      Packets      :      148450735      947295 pps
      Bytes        :      8016344944     409228848 bps
    Transmitted:
      Packets      :      76397439      487512 pps
      Bytes        :     4125461868     210602376 bps
      Tail-dropped packets : Not Available
      RED-dropped packets :      72053285      459783 pps
        Low        :      72053285      459783 pps
        Medium-low  :           0          0 pps
        Medium-high :           0          0 pps
        High        :           0          0 pps
      RED-dropped bytes :     3890877444     198626472 bps
        Low        :     3890877444     198626472 bps
        Medium-low  :           0          0 bps
        Medium-high :           0          0 bps
        High        :           0          0 bps
  Queue: 1, Forwarding classes: expedited-forwarding
    Queued:
      Packets      :           0          0 pps
      Bytes        :           0          0 bps
    Transmitted:
      Packets      :           0          0 pps
      Bytes        :           0          0 bps
      Tail-dropped packets : Not Available
      RED-dropped packets :           0          0 pps
        Low        :           0          0 pps
        Medium-low  :           0          0 pps
        Medium-high :           0          0 pps
        High        :           0          0 pps
      RED-dropped bytes :           0          0 bps
        Low        :           0          0 bps
        Medium-low  :           0          0 bps
        Medium-high :           0          0 bps
        High        :           0          0 bps
  Queue: 2, Forwarding classes: assured-forwarding
    Queued:
      Packets      :      410278257      473940 pps
      Bytes        :     22156199518     204742296 bps
    Transmitted:
      Packets      :      4850003      4033 pps
      Bytes        :     261900162     1742256 bps
      Tail-dropped packets : Not Available
      RED-dropped packets :      405425693      469907 pps
        Low        :      405425693      469907 pps
        Medium-low  :           0          0 pps
        Medium-high :           0          0 pps
        High        :           0          0 pps
      RED-dropped bytes :     21892988124     203000040 bps
        Low        :     21892988124     203000040 bps
        Medium-low  :           0          0 bps
        Medium-high :           0          0 bps
        High        :           0          0 bps
  Queue: 3, Forwarding classes: network-control
    Queued:
      Packets      :           0          0 pps

```

```

Bytes : 0 0 bps
Transmitted:
Packets : 0 0 pps
Bytes : 0 0 bps
Tail-dropped packets : Not Available
RED-dropped packets : 0 0 pps
Low : 0 0 pps
Medium-low : 0 0 pps
Medium-high : 0 0 pps
High : 0 0 pps
RED-dropped bytes : 0 0 bps
Low : 0 0 bps
Medium-low : 0 0 bps
Medium-high : 0 0 bps
High : 0 0 bps
Forwarding classes: 16 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
Queued:
Packets : 76605230 485376 pps
Bytes : 5209211400 264044560 bps
Transmitted:
Packets : 76444631 484336 pps
Bytes : 5198235612 263478800 bps
Tail-dropped packets : Not Available
RED-dropped packets : 160475 1040 pps
Low : 160475 1040 pps
Medium-low : 0 0 pps
Medium-high : 0 0 pps
High : 0 0 pps
RED-dropped bytes : 10912300 565760 bps
Low : 10912300 565760 bps
Medium-low : 0 0 bps
Medium-high : 0 0 bps
High : 0 0 bps
Queue: 1, Forwarding classes: expedited-forwarding
Queued:
Packets : 0 0 pps
Bytes : 0 0 bps
Transmitted:
Packets : 0 0 pps
Bytes : 0 0 bps
Tail-dropped packets : Not Available
RED-dropped packets : 0 0 pps
Low : 0 0 pps
Medium-low : 0 0 pps
Medium-high : 0 0 pps
High : 0 0 pps
RED-dropped bytes : 0 0 bps
Low : 0 0 bps
Medium-low : 0 0 bps
Medium-high : 0 0 bps
High : 0 0 bps
Queue: 2, Forwarding classes: assured-forwarding
Queued:
Packets : 4836136 3912 pps
Bytes : 333402032 2139056 bps
Transmitted:
Packets : 3600866 1459 pps
Bytes : 244858888 793696 bps
Tail-dropped packets : Not Available

```

```

RED-dropped packets :          1225034          2450 pps
  Low                :          1225034          2450 pps
  Medium-low         :              0              0 pps
  Medium-high        :              0              0 pps
  High               :              0              0 pps
RED-dropped bytes   :          83302312        1333072 bps
  Low                :          83302312        1333072 bps
  Medium-low         :              0              0 bps
  Medium-high        :              0              0 bps
  High               :              0              0 bps
Queue: 3, Forwarding classes: network-control
Queued:
  Packets            :              0              0 pps
  Bytes              :              0              0 bps
Transmitted:
  Packets            :              0              0 pps
  Bytes              :              0              0 bps
Tail-dropped packets : Not Available
RED-dropped packets :              0              0 pps
  Low                :              0              0 pps
  Medium-low         :              0              0 pps
  Medium-high        :              0              0 pps
  High               :              0              0 pps
RED-dropped bytes   :              0              0 bps
  Low                :              0              0 bps
  Medium-low         :              0              0 bps
  Medium-high        :              0              0 bps
  High               :              0              0 bps

```

#### Packet Forwarding Engine Chassis Queues:

Queues: 4 supported, 4 in use

Queue: 0, Forwarding classes: best-effort

```

Queued:
  Packets            :          77059796        486384 pps
  Bytes              :          3544750624      178989576 bps
Transmitted:
  Packets            :          77059797        486381 pps
  Bytes              :          3544750670      178988248 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets :              0              0 pps
  Low                :              0              0 pps
  Medium-low         :              0              0 pps
  Medium-high        :              0              0 pps
  High               :              0              0 pps
RED-dropped bytes   :              0              0 bps
  Low                :              0              0 bps
  Medium-low         :              0              0 bps
  Medium-high        :              0              0 bps
  High               :              0              0 bps

```

Queue: 1, Forwarding classes: expedited-forwarding

```

Queued:
  Packets            :              0              0 pps
  Bytes              :              0              0 bps
Transmitted:
  Packets            :              0              0 pps
  Bytes              :              0              0 bps
Tail-dropped packets :              0              0 pps
RED-dropped packets :              0              0 pps
  Low                :              0              0 pps
  Medium-low         :              0              0 pps
  Medium-high        :              0              0 pps

```



```

      High : 0 0 pps
    RED-dropped bytes : 0 0 bps
      Low : 0 0 bps
    Medium-low : 0 0 bps
    Medium-high : 0 0 bps
      High : 0 0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets : 4846580 3934 pps
    Bytes : 222942680 1447768 bps
  Transmitted:
    Packets : 4846580 3934 pps
    Bytes : 222942680 1447768 bps
    Tail-dropped packets : 0 0 pps
    RED-dropped packets : 0 0 pps
      Low : 0 0 pps
    Medium-low : 0 0 pps
    Medium-high : 0 0 pps
      High : 0 0 pps
    RED-dropped bytes : 0 0 bps
      Low : 0 0 bps
    Medium-low : 0 0 bps
    Medium-high : 0 0 bps
      High : 0 0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets : 0 0 pps
    Bytes : 0 0 bps
  Transmitted:
    Packets : 0 0 pps
    Bytes : 0 0 bps
    Tail-dropped packets : 0 0 pps
    RED-dropped packets : 0 0 pps
      Low : 0 0 pps
    Medium-low : 0 0 pps
    Medium-high : 0 0 pps
      High : 0 0 pps
    RED-dropped bytes : 0 0 bps
      Low : 0 0 bps
    Medium-low : 0 0 bps
    Medium-high : 0 0 bps
      High : 0 0 bps

```

#### show interfaces queue (Gigabit Ethernet IQ2 PIC)

```

user@host> show interfaces queue ge-7/1/3
Physical interface: ge-7/1/3, Enabled, Physical link is Up
  Interface index: 170, SNMP ifIndex: 70 Forwarding classes: 16 supported, 4 in
  use Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets : 418390039 10 pps
    Bytes : 38910269752 7440 bps
  Transmitted:
    Packets : 418390039 10 pps
    Bytes : 38910269752 7440 bps
    Tail-dropped packets : Not Available
    RED-dropped packets : 0 0 pps
    RED-dropped bytes : 0 0 bps
Queue: 1, Forwarding classes: expedited-forwarding

```

```

Queued:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps
Transmitted:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :           0           0 pps
  RED-dropped bytes  :           0           0 bps
Queue: 2, Forwarding classes: assured-forwarding
Queued:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps
Transmitted:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :           0           0 pps
  RED-dropped bytes  :           0           0 bps
Queue: 3, Forwarding classes: network-control
Queued:
  Packets      :          7055           1 pps
  Bytes       :         451552          512 bps
Transmitted:
  Packets      :          7055           1 pps
  Bytes       :         451552          512 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :           0           0 pps
  RED-dropped bytes  :           0           0 bps
Forwarding classes: 16 supported, 4 in use Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
Queued:
  Packets      :          1031           0 pps
  Bytes       :         143292           0 bps
Transmitted:
  Packets      :          1031           0 pps
  Bytes       :         143292           0 bps
  Tail-dropped packets : Not Available
  RL-dropped packets  :           0           0 pps
  RL-dropped bytes   :           0           0 bps
  RED-dropped packets :           0           0 pps
  RED-dropped bytes  :           0           0 bps
Queue: 1, Forwarding classes: expedited-forwarding
Queued:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps
Transmitted:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps
  Tail-dropped packets : Not Available
  RL-dropped packets  :           0           0 pps
  RL-dropped bytes   :           0           0 bps
  RED-dropped packets :           0           0 pps
  RED-dropped bytes  :           0           0 bps
Queue: 2, Forwarding classes: assured-forwarding
Queued:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps
Transmitted:
  Packets      :           0           0 pps
  Bytes       :           0           0 bps

```

```

Tail-dropped packets : Not Available
RL-dropped packets   :                0                0 pps
RL-dropped bytes     :                0                0 bps
RED-dropped packets  :                0                0 pps
RED-dropped bytes    :                0                0 bps
Queue: 3, Forwarding classes: network-control
Queued:
Packets              :                77009              11 pps
Bytes                :            6894286              7888 bps
Transmitted:
Packets              :                77009              11 pps
Bytes                :            6894286              7888 bps
Tail-dropped packets : Not Available
RL-dropped packets   :                0                0 pps
RL-dropped bytes     :                0                0 bps
RED-dropped packets  :                0                0 pps
RED-dropped bytes    :                0                0 bps

```

#### Packet Forwarding Engine Chassis Queues:

Queues: 4 supported, 4 in use

Queue: 0, Forwarding classes: best-effort

```

Queued:
Packets              :                1031              0 pps
Bytes                :            147328              0 bps
Transmitted:
Packets              :                1031              0 pps
Bytes                :            147328              0 bps
Tail-dropped packets :                0                0 pps
RED-dropped packets  :                0                0 pps
Low, non-TCP         :                0                0 pps
Low, TCP              :                0                0 pps
High, non-TCP        :                0                0 pps
High, TCP             :                0                0 pps
RED-dropped bytes    :                0                0 bps
Low, non-TCP         :                0                0 bps
Low, TCP              :                0                0 bps
High, non-TCP        :                0                0 bps
High, TCP             :                0                0 bps

```

Queue: 1, Forwarding classes: expedited-forwarding

```

Queued:
Packets              :                0                0 pps
Bytes                :                0                0 bps
Transmitted:
Packets              :                0                0 pps
Bytes                :                0                0 bps
Tail-dropped packets :                0                0 pps
RED-dropped packets  :                0                0 pps
Low, non-TCP         :                0                0 pps
Low, TCP              :                0                0 pps
High, non-TCP        :                0                0 pps
High, TCP             :                0                0 pps
RED-dropped bytes    :                0                0 bps
Low, non-TCP         :                0                0 bps
Low, TCP              :                0                0 bps
High, non-TCP        :                0                0 bps
High, TCP             :                0                0 bps

```

Queue: 2, Forwarding classes: assured-forwarding

```

Queued:
Packets              :                0                0 pps
Bytes                :                0                0 bps
Transmitted:

```

```

Packets          : 0 0 pps
Bytes            : 0 0 bps
Tail-dropped packets : 0 0 pps
RED-dropped packets : 0 0 pps
  Low, non-TCP    : 0 0 pps
  Low, TCP        : 0 0 pps
  High, non-TCP   : 0 0 pps
  High, TCP       : 0 0 pps
RED-dropped bytes : 0 0 bps
  Low, non-TCP    : 0 0 bps
  Low, TCP        : 0 0 bps
  High, non-TCP   : 0 0 bps
  High, TCP       : 0 0 bps
Queue: 3, Forwarding classes: network-control
Queued:
  Packets          : 94386 12 pps
  Bytes            : 13756799 9568 bps
Transmitted:
  Packets          : 94386 12 pps
  Bytes            : 13756799 9568 bps
  Tail-dropped packets : 0 0 pps
  RED-dropped packets : 0 0 pps
    Low, non-TCP    : 0 0 pps
    Low, TCP        : 0 0 pps
    High, non-TCP   : 0 0 pps
    High, TCP       : 0 0 pps
  RED-dropped bytes : 0 0 bps
    Low, non-TCP    : 0 0 bps
    Low, TCP        : 0 0 bps
    High, non-TCP   : 0 0 bps
    High, TCP       : 0 0 bps

```

#### show interfaces queue both-ingress-egress (Gigabit Ethernet IQ2 PIC)

```

user@host> show interfaces queue ge-6/2/0 both-ingress-egress
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
Queued:
  Packets          : Not Available
  Bytes            : 0 0 bps
Transmitted:
  Packets          : 254 0 pps
  Bytes            : 16274 0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets : 0 0 pps
  RED-dropped bytes   : 0 0 bps
Queue: 1, Forwarding classes: expedited-forwarding
Queued:
  Packets          : Not Available
  Bytes            : 0 0 bps
Transmitted:
  Packets          : 0 0 pps
  Bytes            : 0 0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets : 0 0 pps
  RED-dropped bytes   : 0 0 bps
Queue: 2, Forwarding classes: assured-forwarding

```

```

Queued:
  Packets          : Not Available
  Bytes           :                0                0 bps
Transmitted:
  Packets          :                0                0 pps
  Bytes           :                0                0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :                0                0 pps
  RED-dropped bytes  :                0                0 bps
Queue: 3, Forwarding classes: network-control
Queued:
  Packets          : Not Available
  Bytes           :                0                0 bps
Transmitted:
  Packets          :                0                0 pps
  Bytes           :                0                0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :                0                0 pps
  RED-dropped bytes  :                0                0 bps
Forwarding classes: 8 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
Queued:
  Packets          : Not Available
  Bytes           :                0                0 bps
Transmitted:
  Packets          :                3                0 pps
  Bytes           :               126                0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :                0                0 pps
  RED-dropped bytes  :                0                0 bps
Queue: 1, Forwarding classes: expedited-forwarding
Queued:
  Packets          : Not Available
  Bytes           :                0                0 bps
Transmitted:
  Packets          :                0                0 pps
  Bytes           :                0                0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :                0                0 pps
  RED-dropped bytes  :                0                0 bps
Queue: 2, Forwarding classes: assured-forwarding
Queued:
  Packets          : Not Available
  Bytes           :                0                0 bps
Transmitted:
  Packets          :                0                0 pps
  Bytes           :                0                0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :                0                0 pps
  RED-dropped bytes  :                0                0 bps
Queue: 3, Forwarding classes: network-control
Queued:
  Packets          : Not Available
  Bytes           :                0                0 bps
Transmitted:
  Packets          :                0                0 pps
  Bytes           :                0                0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :                0                0 pps
  RED-dropped bytes  :                0                0 bps

```

```

Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets      :      80564692      0 pps
    Bytes        :      3383717100    0 bps
  Transmitted:
    Packets      :      80564692      0 pps
    Bytes        :      3383717100    0 bps
    Tail-dropped packets :      0      0 pps
    RED-dropped packets :      0      0 pps
    RED-dropped bytes  :      0      0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets      :      80564685      0 pps
    Bytes        :      3383716770    0 bps
  Transmitted:
    Packets      :      80564685      0 pps
    Bytes        :      3383716770    0 bps
    Tail-dropped packets :      0      0 pps
    RED-dropped packets :      0      0 pps
    RED-dropped bytes  :      0      0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets      :      0      0 pps
    Bytes        :      0      0 bps
  Transmitted:
    Packets      :      0      0 pps
    Bytes        :      0      0 bps
    Tail-dropped packets :      0      0 pps
    RED-dropped packets :      0      0 pps
    RED-dropped bytes  :      0      0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets      :      9397      0 pps
    Bytes        :      3809052      232 bps
  Transmitted:
    Packets      :      9397      0 pps
    Bytes        :      3809052      232 bps
    Tail-dropped packets :      0      0 pps
    RED-dropped packets :      0      0 pps
    RED-dropped bytes  :      0      0 bps

```

### show interfaces queue ingress (Gigabit Ethernet IQ2 PIC)

```

user@host> show interfaces queue ge-6/2/0 ingress
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets      : Not Available
    Bytes        :      0      0 bps
  Transmitted:
    Packets      :      288      0 pps
    Bytes        :      18450      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets :      0      0 pps
    RED-dropped bytes  :      0      0 bps

```

```

Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets      : Not Available
    Bytes        :                0                0 bps
  Transmitted:
    Packets      :                0                0 pps
    Bytes        :                0                0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets :                0                0 pps
    RED-dropped bytes  :                0                0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets      : Not Available
    Bytes        :                0                0 bps
  Transmitted:
    Packets      :                0                0 pps
    Bytes        :                0                0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets :                0                0 pps
    RED-dropped bytes  :                0                0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets      : Not Available
    Bytes        :                0                0 bps
  Transmitted:
    Packets      :                0                0 pps
    Bytes        :                0                0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets :                0                0 pps
    RED-dropped bytes  :                0                0 bps

```

#### show interfaces queue egress (Gigabit Ethernet IQ2 PIC)

```

user@host> show interfaces queue ge-6/2/0 egress
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets      : Not Available
    Bytes        :                0                0 bps
  Transmitted:
    Packets      :                3                0 pps
    Bytes        :               126                0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets :                0                0 pps
    RED-dropped bytes  :                0                0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets      : Not Available
    Bytes        :                0                0 bps
  Transmitted:
    Packets      :                0                0 pps
    Bytes        :                0                0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets :                0                0 pps
    RED-dropped bytes  :                0                0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:

```

```

Packets          : Not Available
Bytes            :                      0          0 bps
Transmitted:
Packets          :                      0          0 pps
Bytes            :                      0          0 bps
Tail-dropped packets : Not Available
RED-dropped packets :                      0          0 pps
RED-dropped bytes  :                      0          0 bps
Queue: 3, Forwarding classes: network-control
Queued:
Packets          : Not Available
Bytes            :                      0          0 bps
Transmitted:
Packets          :                      0          0 pps
Bytes            :                      0          0 bps
Tail-dropped packets : Not Available
RED-dropped packets :                      0          0 pps
RED-dropped bytes  :                      0          0 bps
Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
Queued:
Packets          :                      80564692      0 pps
Bytes            :                      3383717100      0 bps
Transmitted:
Packets          :                      80564692      0 pps
Bytes            :                      3383717100      0 bps
Tail-dropped packets :                      0          0 pps
RED-dropped packets :                      0          0 pps
RED-dropped bytes  :                      0          0 bps
Queue: 1, Forwarding classes: expedited-forwarding
Queued:
Packets          :                      80564685      0 pps
Bytes            :                      3383716770      0 bps
Transmitted:
Packets          :                      80564685      0 pps
Bytes            :                      3383716770      0 bps
Tail-dropped packets :                      0          0 pps
RED-dropped packets :                      0          0 pps
RED-dropped bytes  :                      0          0 bps
Queue: 2, Forwarding classes: assured-forwarding
Queued:
Packets          :                      0          0 pps
Bytes            :                      0          0 bps
Transmitted:
Packets          :                      0          0 pps
Bytes            :                      0          0 bps
Tail-dropped packets :                      0          0 pps
RED-dropped packets :                      0          0 pps
RED-dropped bytes  :                      0          0 bps
Queue: 3, Forwarding classes: network-control
Queued:
Packets          :                      9538          0 pps
Bytes            :                      3819840         0 bps
Transmitted:
Packets          :                      9538          0 pps
Bytes            :                      3819840         0 bps
Tail-dropped packets :                      0          0 pps
RED-dropped packets :                      0          0 pps
RED-dropped bytes  :                      0          0 bps

```



## show interfaces queue remaining-traffic (Gigabit Ethernet Enhanced DPC)

```

user@host> show interfaces queue ge-2/2/9 remaining-traffic
Physical interface: ge-2/2/9, Enabled, Physical link is Up
  Interface index: 238, SNMP ifIndex: 71
Forwarding classes: 16 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets      :          110208969          472875 pps
    Bytes        :          5951284434        204282000 bps
  Transmitted:
    Packets      :          110208969          472875 pps
    Bytes        :          5951284434        204282000 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :
    Low          :              0              0 pps
    Medium-low   :              0              0 pps
    Medium-high  :              0              0 pps
    High         :              0              0 pps
  RED-dropped bytes :
    Low          :              0              0 bps
    Medium-low   :              0              0 bps
    Medium-high  :              0              0 bps
    High         :              0              0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets      :              0              0 pps
    Bytes        :              0              0 bps
  Transmitted:
    Packets      :              0              0 pps
    Bytes        :              0              0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :
    Low          :              0              0 pps
    Medium-low   :              0              0 pps
    Medium-high  :              0              0 pps
    High         :              0              0 pps
  RED-dropped bytes :
    Low          :              0              0 bps
    Medium-low   :              0              0 bps
    Medium-high  :              0              0 bps
    High         :              0              0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets      :              0              0 pps
    Bytes        :              0              0 bps
  Transmitted:
    Packets      :              0              0 pps
    Bytes        :              0              0 bps
  Tail-dropped packets : Not Available
  RED-dropped packets :
    Low          :              0              0 pps
    Medium-low   :              0              0 pps
    Medium-high  :              0              0 pps
    High         :              0              0 pps
  RED-dropped bytes :
    Low          :              0              0 bps
    Medium-low   :              0              0 bps
    Medium-high  :              0              0 bps

```

```

    High : 0 0 bps
Queue: 3, Forwarding classes: network-control
Queued:
  Packets : 0 0 pps
  Bytes : 0 0 bps
Transmitted:
  Packets : 0 0 pps
  Bytes : 0 0 bps
Tail-dropped packets : Not Available
RED-dropped packets : 0 0 pps
  Low : 0 0 pps
  Medium-low : 0 0 pps
  Medium-high : 0 0 pps
  High : 0 0 pps
RED-dropped bytes : 0 0 bps
  Low : 0 0 bps
  Medium-low : 0 0 bps
  Medium-high : 0 0 bps
  High : 0 0 bps
Forwarding classes: 16 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
Queued:
  Packets : 109355853 471736 pps
  Bytes : 7436199152 256627968 bps
Transmitted:
  Packets : 109355852 471736 pps
  Bytes : 7436198640 256627968 bps
Tail-dropped packets : Not Available
RED-dropped packets : 0 0 pps
  Low : 0 0 pps
  Medium-low : 0 0 pps
  Medium-high : 0 0 pps
  High : 0 0 pps
RED-dropped bytes : 0 0 bps
  Low : 0 0 bps
  Medium-low : 0 0 bps
  Medium-high : 0 0 bps
  High : 0 0 bps
Queue: 1, Forwarding classes: expedited-forwarding
Queued:
  Packets : 0 0 pps
  Bytes : 0 0 bps
Transmitted:
  Packets : 0 0 pps
  Bytes : 0 0 bps
Tail-dropped packets : Not Available
RED-dropped packets : 0 0 pps
  Low : 0 0 pps
  Medium-low : 0 0 pps
  Medium-high : 0 0 pps
  High : 0 0 pps
RED-dropped bytes : 0 0 bps
  Low : 0 0 bps
  Medium-low : 0 0 bps
  Medium-high : 0 0 bps
  High : 0 0 bps
Queue: 2, Forwarding classes: assured-forwarding
Queued:
  Packets : 0 0 pps
  Bytes : 0 0 bps

```

```

Transmitted:
Packets          :                0                0 pps
Bytes            :                0                0 bps
Tail-dropped packets : Not Available
RED-dropped packets :                0                0 pps
  Low            :                0                0 pps
  Medium-low     :                0                0 pps
  Medium-high    :                0                0 pps
  High           :                0                0 pps
RED-dropped bytes  :                0                0 bps
  Low            :                0                0 bps
  Medium-low     :                0                0 bps
  Medium-high    :                0                0 bps
  High           :                0                0 bps
Queue: 3, Forwarding classes: network-control
Queued:
Packets          :                0                0 pps
Bytes            :                0                0 bps
Transmitted:
Packets          :                0                0 pps
Bytes            :                0                0 bps
Tail-dropped packets : Not Available
RED-dropped packets :                0                0 pps
  Low            :                0                0 pps
  Medium-low     :                0                0 pps
  Medium-high    :                0                0 pps
  High           :                0                0 pps
RED-dropped bytes  :                0                0 bps
  Low            :                0                0 bps
  Medium-low     :                0                0 bps
  Medium-high    :                0                0 bps
  High           :                0                0 bps

```

#### show interfaces queue (Channelized OC12 IQE Type 3 PIC in SONET Mode)

```

user@host> show interfaces queue t3-1/1/0:7
Physical interface: t3-1/1/0:7, Enabled, Physical link is Up

  Interface index: 192, SNMP ifIndex: 1948

  Description: full T3 interface connect to 6ce13 t3-3/1/0:7 for FR testing -
  Lam

  Forwarding classes: 16 supported, 9 in use

  Egress queues: 8 supported, 8 in use

  Queue: 0, Forwarding classes: DEFAULT

  Queued:

    Packets          :                214886                13449 pps

    Bytes            :                9884756                5164536 bps

  Transmitted:

    Packets          :                214886                13449 pps

    Bytes            :                9884756                5164536 bps

```

Tail-dropped packets :	0	0 pps
RED-dropped packets :	0	0 pps
Low :	0	0 pps
Medium-low :	0	0 pps
Medium-high :	0	0 pps
High :	0	0 pps
RED-dropped bytes :	0	0 bps
Low :	0	0 bps
Medium-low :	0	0 bps
Medium-high :	0	0 bps
High :	0	0 bps

Queue: 1, Forwarding classes: REALTIME

Queued:

Packets :	0	0 pps
Bytes :	0	0 bps

Transmitted:

Packets :	0	0 pps
Bytes :	0	0 bps
Tail-dropped packets :	0	0 pps
RED-dropped packets :	0	0 pps
Low :	0	0 pps
Medium-low :	0	0 pps
Medium-high :	0	0 pps
High :	0	0 pps
RED-dropped bytes :	0	0 bps
Low :	0	0 bps
Medium-low :	0	0 bps
Medium-high :	0	0 bps
High :	0	0 bps

Queue: 2, Forwarding classes: PRIVATE

## Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

## Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

## Queue: 3, Forwarding classes: CONTROL

## Queued:

Packets	:	60	0 pps
Bytes	:	4560	0 bps

## Transmitted:

Packets	:	60	0 pps
Bytes	:	4560	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps

Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 4, Forwarding classes: CLASS\_B\_OUTPUT

Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 5, Forwarding classes: CLASS\_C\_OUTPUT

Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps

RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 6, Forwarding classes: CLASS\_V\_OUTPUT

Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 7, Forwarding classes: CLASS\_S\_OUTPUT, GETS

Queued:

Packets	:	0	0 pps
---------	---	---	-------

Bytes	:	0	0 bps
Transmitted:			
Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Packet Forwarding Engine Chassis Queues:

Queues: 8 supported, 8 in use

Queue: 0, Forwarding classes: DEFAULT

Queued:

Packets	:	371365	23620 pps
Bytes	:	15597330	7936368 bps

Transmitted:

Packets	:	371365	23620 pps
Bytes	:	15597330	7936368 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps



High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 1, Forwarding classes: REALTIME

Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 2, Forwarding classes: PRIVATE

Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Tail-dropped packets :	0	0 pps
RED-dropped packets :	0	0 pps
Low :	0	0 pps
Medium-low :	0	0 pps
Medium-high :	0	0 pps
High :	0	0 pps
RED-dropped bytes :	0	0 bps
Low :	0	0 bps
Medium-low :	0	0 bps
Medium-high :	0	0 bps
High :	0	0 bps

Queue: 3, Forwarding classes: CONTROL

Queued:

Packets :	32843	0 pps
Bytes :	2641754	56 bps

Transmitted:

Packets :	32843	0 pps
Bytes :	2641754	56 bps
Tail-dropped packets :	0	0 pps
RED-dropped packets :	0	0 pps
Low :	0	0 pps
Medium-low :	0	0 pps
Medium-high :	0	0 pps
High :	0	0 pps
RED-dropped bytes :	0	0 bps
Low :	0	0 bps
Medium-low :	0	0 bps
Medium-high :	0	0 bps
High :	0	0 bps

Queue: 4, Forwarding classes: CLASS\_B\_OUTPUT

## Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

## Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

## Queue: 5, Forwarding classes: CLASS\_C\_OUTPUT

## Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

## Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps

Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 6, Forwarding classes: CLASS\_V\_OUTPUT

Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps
RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

Queue: 7, Forwarding classes: CLASS\_S\_OUTPUT, GETS

Queued:

Packets	:	0	0 pps
Bytes	:	0	0 bps

Transmitted:

Packets	:	0	0 pps
Bytes	:	0	0 bps
Tail-dropped packets	:	0	0 pps

RED-dropped packets	:	0	0 pps
Low	:	0	0 pps
Medium-low	:	0	0 pps
Medium-high	:	0	0 pps
High	:	0	0 pps
RED-dropped bytes	:	0	0 bps
Low	:	0	0 bps
Medium-low	:	0	0 bps
Medium-high	:	0	0 bps
High	:	0	0 bps

### show interfaces queue (QFX Series)

```

user@switch> show interfaces queue xe-0/0/15
Physical interface: xe-0/0/15, Enabled, Physical link is Up
Interface index: 49165, SNMP ifIndex: 539
Forwarding classes: 12 supported, 8 in use
Egress queues: 12 supported, 8 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
  Transmitted:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
    Tail-dropped packets : Not Available
    Total-dropped packets: 0 0 pps
    Total-dropped bytes  : 0 0 bps
Queue: 3, Forwarding classes: fcoe
  Queued:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
  Transmitted:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
    Tail-dropped packets : Not Available
    Total-dropped packets: 0 0 pps
    Total-dropped bytes  : 0 0 bps
0 bps
Queue: 4, Forwarding classes: no-loss
  Queued:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
  Transmitted:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
    Tail-dropped packets : Not Available
    Total-dropped packets: 0 0 pps
    Total-dropped bytes  : 0 0 bps
Queue: 7, Forwarding classes: network-control

```

```

Queued:
  Packets      :          0          0 pps
  Bytes       :          0          0 bps
Transmitted:
  Packets      :          0          0 pps
  Bytes       :          0          0 bps
  Tail-dropped packets : Not Available
  Total-dropped packets:          0          0 pps
  Total-dropped bytes  :          0          0 bps
Queue: 8, Forwarding classes: mcast
Queued:
  Packets      :          0          0 pps
  Bytes       :          0          0 bps
Transmitted:
  Packets      :          0          0 pps
  Bytes       :          0          0 bps
  Tail-dropped packets : Not Available
  Total-dropped packets:          0          0 pps
  Total-dropped bytes  :          0          0 bps

```

#### show interfaces queue l2-statistics (lsq interface)

```

user@switch> show interfaces queue lsq-2/2/0.2 l2-statistics
Logical interface lsq-2/2/0.2 (Index 69) (SNMP ifIndex 1598)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: be
  Queued:
    Packets      :          1          0 pps
    Bytes       :        1001          0 bps
  Transmitted:
    Packets      :          5          0 pps
    Bytes       :        1062          0 bps
    Tail-dropped packets :          0          0 pps
    RED-dropped packets :          0          0 pps
    RED-dropped bytes  :          0          0 bps
Queue: 1, Forwarding classes: ef
  Queued:
    Packets      :          1          0 pps
    Bytes       :        1500          0 bps
  Transmitted:
    Packets      :          6          0 pps
    Bytes       :       1573          0 bps
    Tail-dropped packets :          0          0 pps
    RED-dropped packets :          0          0 pps
    RED-dropped bytes  :          0          0 bps
Queue: 2, Forwarding classes: af
  Queued:
    Packets      :          1          0 pps
    Bytes       :         512          0 bps
  Transmitted:
    Packets      :          3          0 pps
    Bytes       :         549          0 bps
    Tail-dropped packets :          0          0 pps
    RED-dropped packets :          0          0 pps
    RED-dropped bytes  :          0          0 bps
Queue: 3, Forwarding classes: nc
  Queued:
    Packets      :          0          0 pps

```

```

Bytes          : 0 0 bps
Transmitted:
Packets        : 0 0 pps
Bytes          : 0 0 bps
Tail-dropped packets : 0 0 pps
RED-dropped packets : 0 0 pps
RED-dropped bytes  : 0 0 bps
=====

```

### show interfaces queue lsq (lsq-ifd)

```

user@switch> show interfaces queue lsq-1/0/0
Logical interface lsq-1/0/0 (Index 348) (SNMP ifIndex 660)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: be
  Queued:
    Packets      : 55576 1206 pps
    Bytes        : 29622008 5145472 bps
  Transmitted:
    Packets      : 55576 1206 pps
    Bytes        : 29622008 5145472 bps
    Tail-dropped packets : 0 0 pps
    RL-dropped packets  : 0 0 pps
    RL-dropped bytes    : 0 0 bps
    RED-dropped packets : 0 0 pps
    Low               : 0 0 pps
    Medium-low        : 0 0 pps
    Medium-high       : 0 0 pps
    High              : 0 0 pps
    RED-dropped bytes  : 0 0 bps
    Low               : 0 0 bps
    Medium-low        : 0 0 bps
    Medium-high       : 0 0 bps
    High              : 0 0 bps
Queue: 1, Forwarding classes: ef
  Queued:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
  Transmitted:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps
    Tail-dropped packets : 0 0 pps
    RL-dropped packets  : 0 0 pps
    RL-dropped bytes    : 0 0 bps
    RED-dropped packets : 0 0 pps
    Low               : 0 0 pps
    Medium-low        : 0 0 pps
    Medium-high       : 0 0 pps
    High              : 0 0 pps
    RED-dropped bytes  : 0 0 bps
    Low               : 0 0 bps
    Medium-low        : 0 0 bps
    Medium-high       : 0 0 bps
    High              : 0 0 bps
Queue: 2, Forwarding classes: af
  Queued:
    Packets      : 0 0 pps
    Bytes        : 0 0 bps

```

```
Transmitted:
Packets          :          0          0 pps
Bytes            :          0          0 bps
Tail-dropped packets :          0          0 pps
RL-dropped packets :          0          0 pps
RL-dropped bytes  :          0          0 bps
RED-dropped packets :          0          0 pps
  Low            :          0          0 pps
  Medium-low     :          0          0 pps
  Medium-high    :          0          0 pps
  High           :          0          0 pps
RED-dropped bytes :          0          0 bps
  Low            :          0          0 bps
  Medium-low     :          0          0 bps
  Medium-high    :          0          0 bps
  High           :          0          0 bps
Queue: 3, Forwarding classes: nc
Queued:
Packets          :        22231        482 pps
Bytes            :       11849123     2057600 bps
Transmitted:
Packets          :        22231        482 pps
Bytes            :       11849123     2057600 bps
Tail-dropped packets :          0          0 pps
RL-dropped packets :          0          0 pps
RL-dropped bytes  :          0          0 bps
RED-dropped packets :          0          0 pps
  Low            :          0          0 pps
  Medium-low     :          0          0 pps
  Medium-high    :          0          0 pps
  High           :          0          0 pps
RED-dropped bytes :          0          0 bps
  Low            :          0          0 bps
  Medium-low     :          0          0 bps
  Medium-high    :          0          0 bps
  High           :          0          0 bps
```



## CHAPTER 14

# Operational Mode Commands for Layer 2 Learning

- `clear l2-learning mac-move-buffer`
- `show l2-learning global-information`
- `show l2-learning global-mac-count`
- `show l2-learning instance`
- `show l2-learning interface`
- `show l2-learning mac-move-buffer`

## clear l2-learning mac-move-buffer

---

<b>Syntax</b>	<code>clear l2-learning mac-move-buffer</code> <code>&lt;active&gt;</code>
<b>Release Information</b>	Command introduced in Junos OS Release 13.2.
<b>Description</b>	Clear the MAC move buffer entries.
<b>Options</b>	<b>none</b> — Clear the MAC move buffer entries.  <b>active</b> — (Optional) Unblock the interfaces that were blocked by the MAC move action feature. This allows the user to keep the <b>reopen-time</b> configured to a large value, but when the looping error is fixed, the user can manually release the blocking.
<b>Required Privilege Level</b>	clear
<b>List of Sample Output</b>	<a href="#">clear l2-learning mac-move-buffer on page 178</a> <a href="#">clear l2-learning mac-move-buffer active on page 178</a>
<b>Output Fields</b>	When you enter this command, the MAC move buffer entries are deleted.

### Sample Output

#### clear l2-learning mac-move-buffer

```
user@host> clear l2-learning mac-move-buffer
```

#### clear l2-learning mac-move-buffer active

```
user@host> clear l2-learning mac-move-buffer active
```

## show l2-learning global-information

<b>Syntax</b>	<b>show l2-learning global-information</b>
<b>Release Information</b>	Command introduced in Junos OS Release 8.4.
<b>Description</b>	(MX Series routers only) Display Layer 2 learning process-related information for the entire router.
<b>Options</b>	This command has no options.
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show l2-learning global-information on page 179</a>
<b>Output Fields</b>	<a href="#">Table 9 on page 179</a> describes the output fields for the <b>show l2-learning global-information</b> command. Output fields are listed in the approximate order in which they appear.

**Table 9: show l2-learning global-information Output Fields**

Field Name	Field Description
<b>MAC aging interval</b>	Configured timeout interval, in seconds, for all MAC table entries.
<b>MAC learning</b>	Status of MAC learning: <b>Enabled</b> or <b>Disabled</b> .
<b>MAC statistics</b>	Status of MAC accounting: <b>Enabled</b> or <b>Disabled</b> .
<b>MAC limit Count</b>	Configured maximum limit on the number of MAC addresses that can be learned.
<b>MAC limit hit flag</b>	Status of the learned MAC limit hit flag: <b>Enabled</b> (the learned MAC exceeds the global MAC limit) or <b>Disabled</b> (the learned MAC does not exceed the global MAC limit).
<b>MAC packet action drop</b>	Status of action to drop packets after the configured MAC address limit is reached: <b>Enabled</b> (packets are dropped) or <b>Disabled</b> (packets are forwarded).

## Sample Output

### show l2-learning global-information

```

user@host> show l2-learning global-information
Global Configuration:

MAC aging interval      : 300
MAC learning            : Enabled
MAC statistics          : Disabled
MAC limit Count         : 393215
MAC limit hit flag      : Disabled
MAC packet action drop  : Disabled

```



## show l2-learning global-mac-count

---

<b>Syntax</b>	show l2-learning global-mac-count
<b>Release Information</b>	Command introduced in Junos OS Release 9.3.
<b>Description</b>	(MX Series routers only) Display the total number of dynamic and static MAC addresses learned for the entire router.
<b>Options</b>	This command has no options.
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show l2-learning global-mac-count on page 181</a>
<b>Output Fields</b>	Displays the total number of dynamic and static MAC addresses learned for the entire router.

### Sample Output

show l2-learning global-mac-count

```
user@host> show l2-learning global-mac-count
100 dynamic and static MAC addresses learned globally
```

## show l2-learning instance

<b>Syntax</b>	<b>show l2-learning instance</b>
<b>Release Information</b>	(MX Series routers only) Command introduced in Junos OS Release 8.4.
<b>Description</b>	Display Layer 2 learning properties for all the configured routing instances.
<b>Options</b>	This command has no options.
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show l2-learning instance on page 182</a>
<b>Output Fields</b>	<a href="#">Table 10 on page 182</a> describes the output fields for the <b>show l2-learning instance</b> command. Output fields are listed in the approximate order in which they appear.

Table 10: show l2-learning instance Output Fields

Field Name	Field Description
<b>Routing Instance</b>	Name of routing instance.
<b>Bridging Domain</b>	Name of bridging domain.  On MX Series routers you can use the <b>show l2-learning instance &lt;extensive&gt;</b> command option to display the Bridge Service-id information which includes the Config Service ID and the Active Service ID.
<b>Index</b>	Number associated with the routing instance or bridging domain.
<b>Logical System</b>	Name of logical system or <b>Default</b> if no logical system is configured.
<b>Routing instance flags</b>	Status of Layer 2 learning properties for each routing instance: <ul style="list-style-type: none"> <li>• <b>DL</b>—MAC learning is disabled.</li> <li>• <b>SE</b>—MAC accounting is enabled.</li> <li>• <b>AD</b>—Packets are dropped after MAC address limit is reached.</li> <li>• <b>LH</b>—The maximum number of MAC addresses has been learned on the routing instance. The routing instance is not able to learn any additional MAC addresses.</li> </ul>
<b>MAC limit</b>	Maximum number of MAC addresses that can be learned from each interface in the routing instance or bridging domain.

## Sample Output

### show l2-learning instance

```

user@host> show l2-learning instance
Information for routing instance:

Routing Instance flags (DL -disable learning, SE -stats enabled,
```

AD -packet action drop, LH -mac limit hit)

Routing Instance	Bridging Domain	Index	Logical System	Routing flags	MAC limit
__juniper_private1__		1	Default		5000
vs1	vlan100	3	Default		5120
vs1	vlan200	4	Default		5120

## show l2-learning interface

<b>Syntax</b>	<b>show l2-learning interface</b>
<b>Release Information</b>	Command introduced in Junos OS Release 8.4.
<b>Description</b>	(MX Series routers only) Display Layer 2 learning information for all the interfaces.
<b>Options</b>	This command has no options.
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show l2-learning interface on page 184</a>
<b>Output Fields</b>	<a href="#">Table 11 on page 184</a> describes the output fields for the <b>show l2-learning interface</b> command. Output fields are listed in the approximate order in which they appear.

**Table 11: show l2-learning interface Output Fields**

Field Name	Field Description
<b>Logical interface</b>	Name of the logical interface.
<b>Index</b>	Index of the interface.
<b>Routing Instance</b>	Number of the routing instance to which the interface belongs.
<b>Interface device</b>	Value of the order in which the Junos OS finds and initializes the interface.
<b>Logical interface flags</b>	Status of Layer 2 learning properties for each interface: <ul style="list-style-type: none"> <li>• <b>DL</b>—MAC learning is disabled.</li> <li>• <b>SE</b>—MAC accounting is enabled.</li> <li>• <b>AD</b>—Packets are dropped after the MAC interface limit is reached.</li> <li>• <b>MAC limit</b>—Maximum number of MAC addresses that can be learned from the interface.</li> </ul>

## Sample Output

### show l2-learning interface

```
user@host> show l2-learning interface
Information for interface family:
```

```
Logical Interface flags (DL -disable learning, SE -stats enabled,
AD -packet action drop, LH -mac limit hit)
```

Logical interface	Index	Routing instance	Interface device	Logical Interface flags	MAC limit
ge-11/0/3.0	79	3	136		1024
ge-11/1/4.100	84	3	150		1024
ge-11/1/1.100	86	3	147		1024



ge-11/1/0.100	87	3	146	1024
xe-10/2/0.100	88	3	144	1024
xe-10/0/0.100	89	3	129	1024
ge-11/1/0.200	90	4	146	1024
ge-11/1/1.200	91	4	147	1024
ge-11/1/4.200	92	4	150	1024
xe-10/0/0.200	93	4	129	1024
xe-10/2/0.200	94	4	144	1024

## show l2-learning mac-move-buffer

<b>Syntax</b>	<b>show l2-learning mac-move-buffer</b> <b>&lt;brief   detail   extensive&gt;</b> <b>&lt;active&gt;</b>
<b>Release Information</b>	Command introduced in Junos OS Release 9.4.
<b>Description</b>	(MX Series routers only) Display action as a result of configuring the MAC address move feature.
<b>Options</b>	<p><b>none</b>— Display action as a result of the MAC address move feature.</p> <p><b>brief   detail   extensive</b>— (Optional) Display the specified level of output.</p> <p><b>active</b>— (Optional) Display the set of interfaces blocked as a result of the MAC address move action.</p>
<b>Required Privilege Level</b>	view
<b>List of Sample Output</b>	<a href="#">show l2-learning mac-move-buffer active on page 186</a> <a href="#">show l2-learning mac-move-buffer extensive on page 186</a>
<b>Output Fields</b>	Display action as a result of the MAC address move feature.

## Sample Output

### show l2-learning mac-move-buffer active

```

user@host> show l2-learning mac-move-buffer active
MAC Address: 00:00:00:00:01:01, VLAN Id: 0
  Time Rec : 2012-06-25 06:23:41   Bridge Domain: bd10
  Prev IFL : ge-1/0/5.0             New IFL: ge-1/0/6.0
  IFBD    : ge-1/0/6.0:10           Blocked   : YES

```

### show l2-learning mac-move-buffer extensive

```

user@host> show l2-learning mac-move-buffer extensive | display xml
<l2ald-mac-move-buffer>
<l2ald-mac-move-entry junos:style="extensive">
<l2ald-mac-address>aa:00:00:00:02:00</l2ald-mac-address>
<l2ald-learn-vlan-id>0<l2ald-learn-vlan-id>
<l2ald-mac-move-time-rec>11:27:57</l2ald-mac-move-time-rec>
<l2ald-mac-move-bridge-domain>bd</l2ald-mac-move-bridge-domain>
<l2ald-mac-move-from-ifl>ge-1/0/5.200</l2ald-mac-move-from-ifl>
<l2ald-mac-move-to-ifl>ge-1/0/6.200</l2ald-mac-move-to-ifl>
<l2ald-mac-move-to-ifbd>ge-1/0/6.200</l2ald-mac-move-to-ifbd>
<l2ald-mac-move-is-blocked>Yes</l2ald-mac-move-is-blocked>
</l2ald-mac-move-entry>

```

## PART 4

# Index

- [Index on page 189](#)



# Index

## Symbols

#, comments in configuration statements.....	xiv
( ), in syntax descriptions.....	xiv
< >, in syntax descriptions.....	xiv
[ ], in configuration statements.....	xiv
{ }, in configuration statements.....	xiv
(pipe), in syntax descriptions.....	xiv

## B

bandwidth statement.....	102
usage guidelines.....	47
braces, in configuration statements.....	xiv
brackets	
angle, in syntax descriptions.....	xiv
square, in configuration statements.....	xiv
bridge domain	
dual VLAN tags.....	84
Layer 2, displaying.....	121
routing interface.....	78
service identifier.....	79
VLAN identifier.....	81
VLAN identifier list.....	11, 19
bridge-domains statement.....	61
bridge-options statement.....	62

## C

clear bridge mac-table command.....	118
clear interfaces mac-database command.....	119
clear interfaces mac-database statistics	
command.....	120
comments, in configuration statements.....	xiv
conventions	
text and syntax.....	xiii
CoS	
byte count by interface hardware	
displaying.....	144
curly braces, in configuration statements.....	xiv
customer support.....	xv
contacting JTAC.....	xv

## D

documentation	
comments on.....	xv
domain-type statement.....	63

## F

flooding	
Layer 2 bridging, displaying.....	123
font conventions.....	xiii

## G

global-mac-limit statement.....	107
usage guidelines.....	51
global-mac-statistics statement.....	107
usage guidelines.....	50
global-mac-table-aging-time statement.....	108
usage guidelines.....	49
global-no-mac-learning statement.....	108
usage guidelines.....	52

## I

interface learning information	
Layer 2, displaying.....	184
interface statement	
bridge domain.....	64
virtual switch.....	64
interface-mac-limit statement.....	65, 87, 109
set of bridge domains	
usage guidelines.....	43
trunk port	
usage guidelines.....	43
usage guidelines.....	40
irb	
and mirroring.....	72

## L

l2-learning statement.....	106
Layer 2	
switching example.....	12
Layer 2 bridging	
bridge domain, displaying.....	121
flooding, displaying.....	123
interface learning information, displaying.....	184
learning information, displaying.....	179, 181
learning properties, displaying.....	182
MAC address table, clearing.....	118
MAC address table, displaying.....	130
statistics, displaying.....	135

learning information		
Layer 2, displaying.....	179, 181	
learning properties		
Layer 2, displaying.....	182	
<b>M</b>		
MAC address table		
Layer 2, clearing address information.....	118	
Layer 2, displaying.....	130	
mac-statistics statement.....	67, 89	
set of bridge domains		
usage guidelines.....	45	
trunk port		
usage guidelines.....	45	
usage guidelines.....	41	
mac-table-aging-time statement.....	71	
mac-table-size statement.....	69, 91	
set of bridge domains		
usage guidelines.....	44	
trunk port		
usage guidelines.....	44	
usage guidelines.....	39	
manuals		
comments on.....	xv	
mirroring		
and irb.....	72	
<b>N</b>		
no-irb-layer-2-copy statement.....	72	
no-mac-learning statement.....	73, 93	
set of bridge domains		
usage guidelines.....	43	
trunk port		
usage guidelines.....	43	
usage guidelines.....	37	
<b>P</b>		
packet-action statement.....	75, 95, 111	
parentheses, in syntax descriptions.....	xiv	
<b>R</b>		
routing instances for VPLS		
VLAN identifier list.....	19	
routing-interface statement.....	78	
<b>S</b>		
service-id statement.....	79	
show bridge domain command.....	121	
show bridge flood command.....	123	
show bridge mac-table command.....	130	
show bridge statistics command.....	135	
show interfaces queue command.....	138	
show l2-learning global-information		
command.....	179	
show l2-learning global-mac-count command.....	181	
show l2-learning instance command.....	182	
show l2-learning interface command.....	184	
static-mac statement.....	80	
usage guidelines.....	38	
statistics		
Layer 2 bridging, displaying.....	135	
support, technical See technical support		
switch-options statement.....	86	
switching		
example.....	12	
syntax conventions.....	xiii	
<b>T</b>		
technical support		
contacting JTAC.....	xv	
tunnel interfaces		
configuring, MX Series routers.....	47	
tunnel-services statement.....	100	
usage guidelines.....	47	
<b>V</b>		
virtual switch		
configuring.....	4, 33	
dual VLAN tags.....	84	
routing interface.....	78	
VLAN identifier.....	81	
with VPLS ports.....	30	
virtual-switch statement		
usage guidelines.....	33	
VLAN identifiers		
configuring.....	19	
vlan-id statement.....	81	
vlan-id-list		
usage guidelines.....	11	
vlan-id-list statement.....	83	
vlan-tags statement.....	84	
VPLS ports in a virtual switch.....	30	