# Junos® OS

## Ethernet Switching and Layer 2 Transparent Mode Feature Guide for Security Devices

Modified: 2018-03-08

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net



*Junos® OS Ethernet Switching and Layer 2 Transparent Mode Feature Guide for Security Devices*


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 https://www.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# Table of Contents

# List of Figures

# List of Tables

# About the Documentation



## Documentation and Release Notes

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

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 https://www.juniper.net/books.

## Supported Platforms

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

- SRX Series
- vSRX

## 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.xsl;
       }
     }
   }
   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.xsl; }
   ```

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 CLI Explorer.

## Documentation Conventions

Table 1 on page xix 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 xx defines the text and syntax conventions used in this guide.

## Table 2: Text and Syntax Conventions

| Convention | Description | Examples |
|---|---|---|
| **Bold text like this** | Represents text that you type. | To enter configuration mode, type the **configure** command:<br><br>user@host> **configure** |
| `Fixed-width text like this` | Represents output that appears on the terminal screen. | `user@host>` **show chassis alarms**<br><br>`No alarms currently active` |
| *Italic text like this* | • Introduces or emphasizes important new terms.<br>• Identifies guide names.<br>• Identifies RFC and Internet draft titles. | • A policy *term* is a named structure that defines match conditions and actions.<br>• *Junos OS CLI User Guide*<br>• RFC 1997, *BGP Communities Attribute* |
| *Italic text like this* | Represents variables (options for which you substitute a value) in commands or configuration statements. | Configure the machine's domain name:<br><br>[edit]<br>root@# **set system domain-name** *domain-name* |
| **Text like this** | Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components. | • To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level.<br>• The console port is labeled **CONSOLE**. |
| < > (angle brackets) | Encloses optional keywords or variables. | **stub <default-metric** *metric***>;** |
| \| (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. | **broadcast \| multicast**<br><br>**(***string1* **\|** *string2* **\|** *string3***)** |
| # (pound sign) | Indicates a comment specified on the same line as the configuration statement to which it applies. | **rsvp { # Required for dynamic MPLS only** |
| [ ] (square brackets) | Encloses a variable for which you can substitute one or more values. | **community name members [** *community-ids* **]** |
| Indention and braces ( { } ) | Identifies a level in the configuration hierarchy. | [edit]<br>routing-options {<br>  static {<br>  route default {<br>  nexthop *address*;<br>  retain;<br>  }<br>  }<br>} |
| ; (semicolon) | Identifies a leaf statement at a configuration hierarchy level. | |

### GUI Conventions

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

| Convention | Description | Examples |
|---|---|---|
| **Bold text like this** | Represents graphical user interface (GUI) items you click or select. | • In the Logical Interfaces box, select **All Interfaces**.<br>• To cancel the configuration, click **Cancel**. |
| > (bold right angle bracket) | Separates levels in a hierarchy of menu selections. | In the configuration editor hierarchy, select **Protocols>Ospf**. |

## 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 of the Juniper Networks TechLibrary site at https://www.juniper.net/documentation/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/documentation/feedback/.
- E-mail—Send your comments to 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 Partner Support Service 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 https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit https://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: https://www.juniper.net/customers/support/
- Search for known bugs: https://prsearch.juniper.net/
- Find product documentation: https://www.juniper.net/documentation/
- Find solutions and answer questions using our Knowledge Base: https://kb.juniper.net/
- Download the latest versions of software and review release notes: https://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: https://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: https://www.juniper.net/cm/

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

## 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 https://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 https://www.juniper.net/support/requesting-support.html.

PART 1

# Overview



CHAPTER 1

# Introduction to Switching and Layer 2 Transparent Mode



## Ethernet Switching and Layer 2 Transparent Mode Overview

**Supported Platforms** SRX Series, vSRX

Layer 2 transparent mode provides the ability to deploy the firewall without making changes to the existing routing infrastructure. The firewall is deployed as a Layer 2 switch with multiple VLAN segments and provides security services within VLAN segments. Secure wire is a special version of Layer 2 transparent mode that allows bump-in-wire deployment.

Ethernet switching forwards the Ethernet frames within or across the LAN segment (or VLAN) using the Ethernet MAC address information. Ethernet switching on the SRX1500 device is performed in the hardware using ASICs.

Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, use the **set protocols l2-learning global-mode(transparent-bridge | switching)** command to switch between the Layer 2 transparent bridge mode and Ethernet switching mode. After switching the mode, you must reboot the device for the configuration to take effect.

NOTE: On SRX1500, the default Layer 2 global mode is transparent-bridgemode.

NOTE: Starting in Junos OS Release 15.1X49-D50 and Junos OS Release 17.3R1, the factory-default configuration of the SRX300, SRX320, SRX340, and SRX345 devices is switching mode. When these devices are loaded or reset with the factory-default configuration, they start up in switching mode.

NOTE: Starting with Junos OS Release 15.1X49-D100, on SRX300, SRX320, SRX340, SRX345, SRX550, and SRX550M devices, when Layer 2 global mode configuration is deleted, the default Layer 2 global mode configuration is changed from transparent-bridge to switching mode. You must explicitly configure Layer 2 transparent-bridge mode for the SRX300, SRX320, SRX340, SRX345, SRX550, and SRX550 devices that work in transparent mode. Use the command **set protocols l2-learning global-mode transparent-bridge** before rebooting the devices with Junos OS 15.1X49-D100 image.

The Layer 2 protocol supported in switching mode is Link Aggregation Control Protocol (LACP).

You can configure Layer 2 transparent mode on a redundant Ethernet interface. Use the following commands to define a redundant Ethernet interface:

- **set interfaces *interface-name* ether-options redundant-parent *reth-interface-name***
- **set interfaces *reth-interface-name* redundant-ether-options redundancy-group *number***

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D40 | Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, use the **set protocols l2-learning global-mode(transparent-bridge \| switching)** command to switch between the Layer 2 transparent bridge mode and Ethernet switching mode. |
| 15.1X49-D100 | Starting with Junos OS Release 15.1X49-D100, on SRX300, SRX320, SRX340, SRX345, SRX550, and SRX550M devices, when Layer 2 global mode configuration is deleted, the default Layer 2 global mode configuration is changed from transparent-bridge to switching mode. |

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- **global-mode (Protocols) on page 339**
- **l2-learning (Protocols) on page 348**

CHAPTER 2

# Configuring Interfaces



## Understanding Layer 2 Interfaces on Security Devices

**Supported Platforms** SRX Series, vSRX

Layer 2 logical interfaces are created by defining one or more logical units on a physical interface with the family address type **ethernet-switching**. If a physical interface has a **ethernet-switching** family logical interface, it cannot have any other family type in its logical interfaces. A logical interface can be configured in one of the following modes:

- Access mode—Interface accepts untagged packets, assigns the specified VLAN identifier to the packet, and forwards the packet within the VLAN that is configured with the matching VLAN identifier.
- Trunk mode—Interface accepts any packet tagged with a VLAN identifier that matches a specified list of VLAN identifiers. Trunk mode interfaces are generally used to interconnect switches. To configure a VLAN identifier for untagged packets received on the physical interface, use the **native-vlan-id** option. If the **native-vlan-id** option is not configured, untagged packets are dropped.

NOTE: **Multiple trunk mode logical interfaces can be defined, as long as the VLAN identifiers of a trunk interface do not overlap with those of another trunk interface. The native-vlan-id must belong to a VLAN identifier list configured for a trunk interface.**

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Example: Configuring Layer 2 Logical Interfaces on Security Devices on page 6
- Understanding Transparent Mode Conditions on Security Devices on page 28

## Example: Configuring Layer 2 Logical Interfaces on Security Devices

**Supported Platforms** SRX Series, vSRX

This example shows how to configure a Layer 2 logical interface as a trunk port so that the incoming packets can be selectively redirected to a firewall or other security device.



### Requirements

Before you begin, configure the VLANs. See "Example: Configuring VLANs on Security Devices" on page 31.

### Overview

In this example, you configure logical interface ge-3/0/0.0 as a trunk port that carries traffic for packets tagged with VLAN identifiers 1 through 10; this interface is implicitly assigned to the previously configured VLANs vlan-a and vlan-b. Then you assign a VLAN ID of 10 to any untagged packets received on physical interface ge-3/0/0.

### 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set unit 0 family ethernet-switching interface-mode trunk vlan members 1–10
set vlan-tagging native-vlan-id 10
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a Layer 2 logical interface as a trunk port:

1. Configure the logical interface.

   [edit interfaces ge-3/0/0]

```
user@host# set unit 0 family ethernet-switching interface-mode trunk vlan members
  1–10
```

2. Specify a VLAN ID for untagged packets.

```
[edit interfaces ge-3/0/0]
user@host# set vlan-tagging native-vlan-id 10
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

### Verification

To verify the configuration is working properly, enter the **show interfaces ge-3/0/0** and **show interfaces ge-3/0/0.0** commands.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Layer 2 Interfaces on Security Devices on page 5
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Example: Configuring Layer 2 Security Zones on page 38

## Understanding Integrated Routing and Bridging Interfaces on a Security Device

**Supported Platforms** SRX Series, vSRX

For VLANs configured with a single VLAN identifier, you can optionally configure an integrated routing and bridging (IRB) interface for management traffic in the VLAN. An IRB interface acts as a Layer 3 routing interface for a VLAN.

NOTE: If you specify a VLAN identifier list in the VLAN configuration, you cannot configure an IRB interface for the VLAN.

Packets arriving on a Layer 2 interface that are destined for the device's MAC address are classified as Layer 3 traffic while packets that are not destined for the device's MAC address are classified as Layer 2 traffic. Packets destined for the device's MAC address are sent to the IRB interface. Packets from the device's routing engine are sent out the IRB interface.

You create an IRB logical interface in a similar manner as a Layer 3 interface, but the IRB interface does not support traffic forwarding or routing. The IRB interface cannot be assigned to a security zone; however, you can configure certain services on a per-zone basis to allow host-inbound traffic for management of the device. This allows you to control the type of traffic that can reach the device from interfaces bound to a specific zone.

NOTE:

- On SRX1400, SRX1500, SRX3400, SRX3600, SRX4100, SRX4200, SRX4600, SRX5600, and SRX5800 devices, we support an IRB interface that allows you to terminate management connections in transparent mode. However, you cannot route traffic on that interface or terminate IPsec VPNs. (Platform support depends on the Junos OS release in your installation.)
- You can configure only one IRB logical interface for each VLAN.

NOTE: On SRX300, SRX320, SRX340, SRX345 devices, and SRX550M on the IRB interface, the following features are not supported:

- IS-IS (family ISO)
- Encapsulations (Ether CCC, VLAN CCC, VPLS, PPPoE, and so on) on VLAN interfaces
- CLNS
- DVMRP
- VLAN interface MAC change
- G-ARP
- Change VLAN-Id for VLAN interface

NOTE: Starting with Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, interface statistics are supported on the IRB logical interface for SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

To verify the IRB logical interface statistics, enter the **show interfaces irb.<index> extensive** and **show interfaces irb.<index>statistics** commands.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Example: Configuring an IRB Interface on a Security Device on page 8
- Understanding VLANs on Security Devices on page 29
- Example: Configuring VLANs on Security Devices on page 31

## Example: Configuring an IRB Interface on a Security Device

**Supported Platforms** SRX Series, vSRX

This example shows how to configure an IRB interface so it can act as a Layer 3 routing interface for a VLAN.



## Requirements

Before you begin, configure a VLAN with a single VLAN identifier. See "Example: Configuring VLANs on Security Devices" on page 31.

## Overview

In this example, you configure the IRB logical interface unit 0 with the family type inet and IP address 10.1.1.1/24, and then reference the IRB interface irb.10 in the vlan10 configuration. Then you enable Web authentication on the IRB interface and activate the webserver on the device.

NOTE: To complete the Web authentication configuration, you must perform the following tasks:

- Define the access profile and password for a Web authentication client.
- Define the security policy that enables Web authentication for the client.

Either a local database or an external authentication server can be used as the Web authentication server.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-1/0/0 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-1/0/0 unit 0 family ethernet-switching vlan members 10
set interface irb unit 0 family inet address 10.1.1.1/24 web-authentication http
set vlans vlan10 vlan-id 10
set vlans vlan10 l3-interface irb.10
set system services web-management http
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure an IRB interface:

1. Create a Layer 2 trunk interface.

```
[edit]
user@host# set interfaces ge-1/0/0 unit 0 family ethernet-switching interface-mode
  trunk
user@host# set interfaces ge-1/0/0 unit 0 family ethernet-switching vlan members
  10
```

2. Create an IRB logical interface.

```
[edit]
user@host# set interface irb unit 0 family inet address 10.1.1.1/24 web-authentication
  http
```

3. Create a Layer 2 VLAN.

```
[edit]
user@host# set vlans vlan10 vlan-id 10
```

4. Associate the IRB interface with the VLAN.

```
[edit]
user@host# set vlans vlan10 l3-interface irb.10
```

5. Activate the webserver.

```
[edit]
user@host# set system services web-management http
```

6. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

## Verification

To verify the configuration is working properly, enter the **show interface irb** , and **show vlans** commands.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Integrated Routing and Bridging Interfaces on a Security Device on page 7
- Example: Configuring Layer 2 Security Zones on page 38
- Understanding VLANs on Security Devices on page 29

## Understanding Mixed Mode (Transparent and Route Mode) on Security Devices

**Supported Platforms** SRX Series, vSRX

Mixed mode supports both transparent mode (Layer 2) and route mode (Layer 3); it is the default mode. You can configure both Layer 2 and Layer 3 interfaces simultaneously using separate security zones.

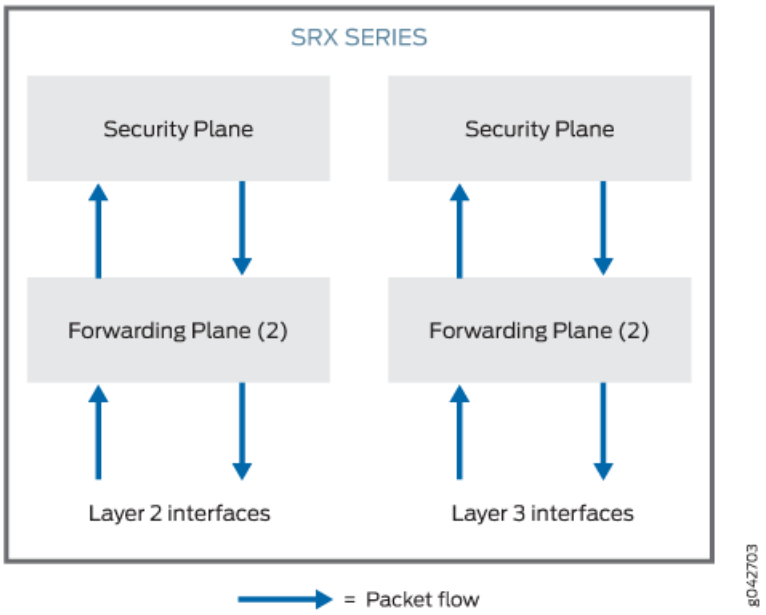
NOTE: **For the mixed mode configuration, you must reboot the device after you commit the changes. However, for SRX5000 line devices, reboot is not required.**

In mixed mode (Transparent and Route Mode):

- There is no routing among IRB interfaces and between IRB interfaces and Layer 3 interfaces.
- The user logical system is not supported for Layer 2 traffic. However, you can configure Layer 2 traffic using the root logical system.
- You can configure Layer 3 interfaces using both the user logical system and the root logical system.

The device in Figure 1 on page 12 looks like two separate devices. One device runs in Layer 2 transparent mode and the other device runs in Layer 3 routing mode. But both devices run independently. Packets cannot be transferred between the Layer 2 and Layer 3 interfaces, because there is no routing among IRB interfaces and between IRB interfaces and Layer 3 interfaces.

Figure 1: Architecture of Mixed Transparent and Route Mode




In mixed mode, the Ethernet physical interface can be either a Layer 2 interface or a Layer 3 interface, but the Ethernet physical interface cannot be both simultaneously. However, Layer 2 and Layer 3 families can exist on separate physical interfaces on the same device.

Table 3 on page 12 lists the Ethernet physical interface types and supported family types.

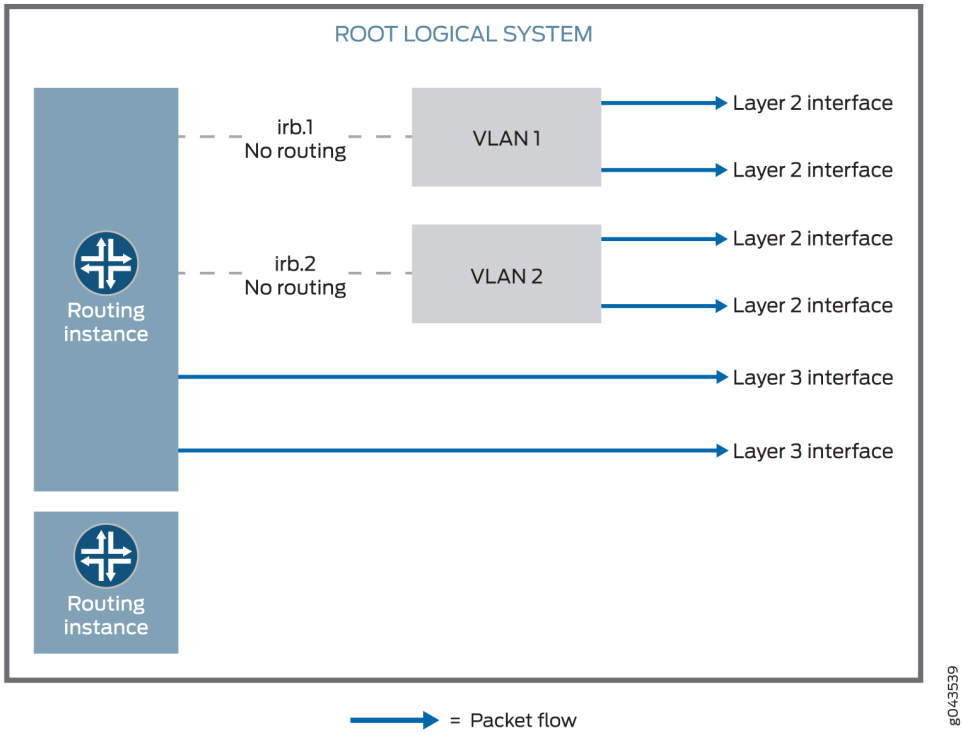
Table 3: Ethernet Physical Interface and Supported Family Types

| Ethernet Physical Interface Type | Supported Family Type |
|---|---|
| Layer 2 Interface | **ethernet-switching** |
| Layer 3 Interface | **inet** and **inet6** |

NOTE: Multiple routing instances are supported.

You can configure both the pseudointerface **irb.x** and the Layer 3 interface under the same default routing instance using either a default routing instance or a user-defined routing instance. See Figure 2 on page 13.

Figure 2: Mixed Transparent and Route Mode




Packets from the Layer 2 interface are switched within the same VLAN, or they connect to the host through the IRB interface. Packets cannot be routed to another IRB interface or a Layer 3 interface through their own IRB interface.

Packets from the Layer 3 interface are routed to another Layer 3 interface. Packets cannot be routed to a Layer 2 interface through an IRB interface.

Table 4 on page 13 lists the security features that are supported in mixed mode and the features that are not supported in transparent mode for Layer 2 switching.

Table 4: Security Features Supported in Mixed Mode (Transparent and Route Mode)

| Mode Type | Supported | Not Supported |
|---|---|---|
| Mixed mode | • Application Layer Gateways (ALGs)<br>• Firewall User Authentication (FWAUTH)<br>• Intrusion Detection and Prevention (IDP)<br>• Screen<br>• AppSecure | • Unified Threat Management (UTM) |

**Table 4: Security Features Supported in Mixed Mode (Transparent and Route Mode) *(continued)***

| Mode Type | Supported | Not Supported |
|---|---|---|
| Route mode (Layer 3 interface) | • Network Address Translation (NAT)<br>• VPN | — |
| Transparent mode (Layer 2 interface) | | • Network Address Translation (NAT)<br>• VPN<br>• Unified Threat Management (UTM) |

Starting in Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, some conditions apply to mixed-mode operations. Note the conditions here:

- On SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices, you cannot configure Ethernet switching and virtual private LAN service (VPLS) using mixed mode (Layer 2 and Layer 3).
- On SRX5400, SRX5600, and SRX5800 devices, you do not have to reboot the device when you configure VLAN.

**Release History Table**

| Release | Description |
|---|---|
| 12.3X48-D10 | Starting in Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, some conditions apply to mixed-mode operations. |

**Related Documentation**

- Example: Improving Security Services by Configuring an SRX Series Device Using Mixed Mode (Transparent and Route Mode) on page 14
- Understanding Secure Wire on Security Devices on page 75

## Example: Improving Security Services by Configuring an SRX Series Device Using Mixed Mode (Transparent and Route Mode)

**Supported Platforms** SRX Series, vSRX

You can configure an SRX Series device using both transparent mode (Layer 2) and route mode (Layer 3) simultaneously to simplify deployments and to improve security services.

This example shows how to pass the Layer 2 traffic from interface ge-0/0/1.0 to interface ge-0/0/0.0 and Layer 3 traffic from interface ge-0/0/2.0 to interface ge-0/0/3.0.

- Requirements on page 15
- Overview on page 15

## Requirements

This example uses the following hardware and software components:

- An SRX Series device
- Four PCs

Before you begin:

- Create a separate security zone for Layer 2 and Layer 3 interfaces. See "Understanding Layer 2 Security Zones" on page 37.

## Overview

In enterprises where different business groups have either Layer 2 or Layer 3 based security solutions, using a single mixed mode configuration simplifies their deployments. In a mixed mode configuration, you can also provide security services with integrated switching and routing.

In addition, you can configure an SRX Series device in both standalone and chassis cluster mode using mixed mode.

In mixed mode (default mode), you can configure both Layer 2 and Layer 3 interfaces simultaneously using separate security zones.

NOTE: **For the mixed mode configuration, you must reboot the device after you commit the changes. However, for SRX5000 line devices, reboot is not required.**

In this example, first you configure a Layer 2 family type called Ethernet switching to identify Layer 2 interfaces. You set the IP address 10.10.10.1/24 to IRB interface. Then you create zone L2 and add Layer 2 interfaces ge-0/0/1.0 and ge-0/0/0.0 to it.

Next you configure a Layer 3 family type inet to identify Layer 3 interfaces. You set the IP address 192.0.2.1/24 to interface ge-0/0/2.0 and the IP address 192.0.2.3/24 to interface ge-0/0/3. Then you create zone L3 and add Layer 3 interfaces ge-0/0/2.0 and ge-0/0/3.0 to it.

## Topology

Figure 3 on page 16 shows a mixed mode topology.

### Figure 3: Mixed Mode Topology

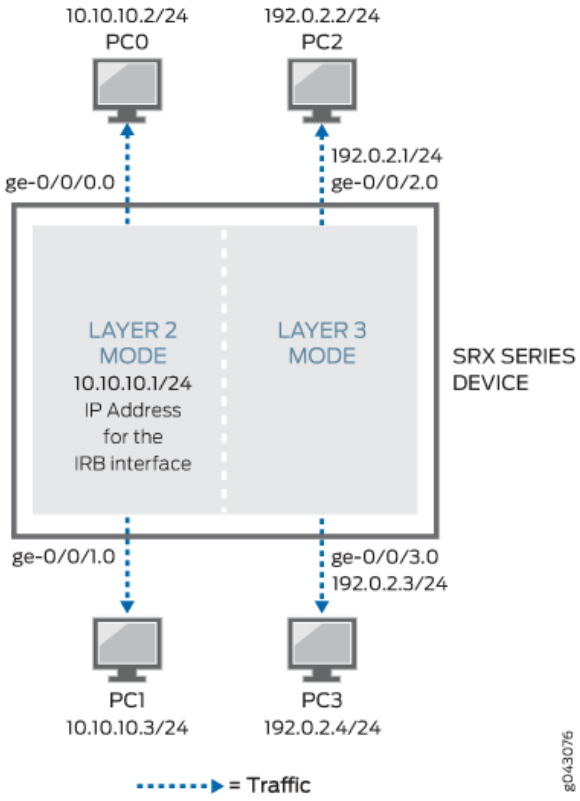



Table 5 on page 16 shows the parameters configured in this example.

### Table 5: Layer 2 and Layer 3 Parameters

| Parameter | Description |
|---|---|
| L2 | Layer 2 zone. |
| ge-0/0/1.0 and ge-0/0/0.0 | Layer 2 interfaces added to the Layer 2 zone. |
| L3 | Layer 3 zone. |
| ge-0/0/2.0 and ge-0/0/3.0 | Layer 3 interfaces added to the Layer 3 zone. |
| 10.10.10.1/24 | IP address for the IRB interface. |
| 192.0.2.1/24 and 192.0.2.3/24 | IP addresses for the Layer 3 interface. |

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/0 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/0 unit 0 family ethernet-switching vlan members 10
set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members 10
set interfaces irb unit 10 family inet address 10.10.10.1/24
set security zones security-zone L2 interfaces ge-0/0/1.0
set security zones security-zone L2 interfaces ge-0/0/0.0
set vlans vlan-10 vlan-id 10
set vlans vlan-10 l3-interface irb.10
set interfaces ge-0/0/2 unit 0 family inet address 192.0.2.1/24
set interfaces ge-0/0/3 unit 0 family inet address 192.0.2.3/24
set security policies default-policy permit-all
set security zones security-zone L2 host-inbound-traffic system-services any-service
set security zones security-zone L2 host-inbound-traffic protocols all
set security zones security-zone L3 host-inbound-traffic system-services any-service
set security zones security-zone L3 host-inbound-traffic protocols all
set security zones security-zone L3 interfaces ge-0/0/2.0
set security zones security-zone L3 interfaces ge-0/0/3.0
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Layer 2 and Layer 3 interfaces:

1. Create a Layer 2 family type to configure Layer 2 interfaces.

   ```
   [edit interfaces]
   user@host# set ge-0/0/0 unit 0 family ethernet-switching interface-mode access
   user@host# set ge-0/0/0 unit 0 family ethernet-switching vlan members 10
   user@host# set ge-0/0/1 unit 0 family ethernet-switching interface-mode access
   user@host# set ge-0/0/1 unit 0 family ethernet-switching vlan members 10
   ```

2. Configure an IP address for the IRB interface.

   ```
   [edit interfaces]
   user@host# set irb unit 10 family inet address 10.10.10.1/24
   ```

3. Configure Layer 2 interfaces.

   ```
   [edit security zones security-zone L2 interfaces]
   user@host# set ge-0/0/1.0
   user@host#  set ge-0/0/0.0
   ```

4. Configure VLAN.

```
[edit vlans vlan-10]
user@host# set vlan-id 10
user@host# set l3-interface irb.10
```

5. Configure IP addresses for Layer 3 interfaces.

```
[edit interfaces]
user@host# set ge-0/0/2.0 unit 0 family inet address 192.0.2.1/24
user@host# set ge-0/0/3.0 unit 0 family inet address 192.0.2.3/24
```

6. Configure the policy to permit the traffic.

```
[edit security policies]
user@host# set default-policy permit-all
```

7. Configure Layer 3 interfaces.

```
[edit security zones security-zone]
user@host#  set L2 host-inbound-traffic system-services any-service
user@host#  set L2 host-inbound-traffic protocols all
user@host#  set L3 host-inbound-traffic system-services any-service
user@host# set L3 host-inbound-traffic protocols all
user@host#  set L3 interfaces ge-0/0/2.0
user@host# set L3 interfaces ge-0/0/3.0
```

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

```
[edit]
user@host# show interfaces
  ge-0/0/0 {
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  ge-0/0/1 {
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  ge-0/0/2 {
    unit 0 {
      family inet {
        address 192.0.2.1/24;
```

```
        }
      }
    }
    ge-0/0/3 {
      unit 0 {
        family inet {
          address 192.0.2.2/24;
        }
      }
    }
    irb {
      unit 10 {
        family inet {
          address 10.10.10.1/24;
        }
      }
    }
[edit]
user@host#  show security policies
default-policy {
  permit-all;
}
[edit]
user@host# show vlans
  vlan-10 {
    vlan-id 10;
    l3-interface irb.10;
  }
[edit]
user@host#  show security zones
  security-zone L2 {
    host-inbound-traffic {
      system-services {
        any-service;
      }
      protocols {
        all;
      }
    }
    interfaces {
      ge-0/0/1.0;
      ge-0/0/0.0;
    }
  }
  security-zone L3 {
    host-inbound-traffic {
      system-services {
        any-service;
      }
      protocols {
        all;
      }
    }
    interfaces {
      ge-0/0/2.0;
      ge-0/0/3.0;
```

```
    }
  }
```

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

## Verification

Confirm that the configuration is working properly.


- Verifying the Layer 2 and Layer 3 Interfaces and Zones on page 20
- Verifying the Layer 2 and Layer 3 Session on page 21


### Verifying the Layer 2 and Layer 3 Interfaces and Zones

**Purpose** Verify that the Layer 2 and Layer 3 interfaces and Layer 2 and Layer 3 zones are created.

**Action** From operational mode, enter the **show security zones** command.

```
user@host> show security zones
 Security zone: HOST
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

Security zone: L2
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 2
  Interfaces:
    ge-0/0/0.0
    ge-0/0/1.0

Security zone: L3
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 2
  Interfaces:
    ge-0/0/2.0
    ge-0/0/3.0

Security zone: junos-host
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

**Meaning** The output shows the Layer 2 (L2) and Layer 3 (L3) zone names and the number and names of Layer 2 and Layer 3 interfaces bound to the L2 and L3 zones.

### Verifying the Layer 2 and Layer 3 Session

**Purpose** Verify that the Layer 2 and Layer 3 sessions are established on the device.

**Action** From operational mode, enter the **show security flow session** command.

```
user@host> show security flow session
Session ID: 130000050, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 10.10.10.2/22 --> 10.10.10.3/28;icmp, If: ge-0/0/0.0, Pkts: 1, Bytes: 98
  Out: 10.10.10.3/245 --> 10.10.10.2/248;icmp, If: ge-0/0/1.0, Pkts: 1, Bytes:
98

Session ID: 130000051, Policy name: default-policy-02/2, Timeout: 4, Valid
  In: 192.0.2.1/17 --> 192.0.2.2/19;icmp, If: ge-0/0/2.0, Pkts: 1, Bytes: 84
Out: 192.0.2.2/212 --> 192.0.2.1/218;icmp, If: ge-0/0/3.0, Pkts: 1, Bytes: 84
```

**Meaning** The output shows active sessions on the device and each session's associated security policy.

- **Session ID 130000050**—Number that identifies the Layer 2 session. Use this ID to get more information about the Layer 2 session such as policy name or number of packets in and out.
- **default-policy-00/2**—Default policy name that permitted the Layer 2 traffic.
- **In**—Incoming flow (source and destination Layer 2 IP addresses with their respective source and destination port numbers, session is ICMP, and the source interface for this session is ge-0/0/0.0).
- **Out**—Reverse flow (source and destination Layer 2 IP addresses with their respective source and destination port numbers, session is ICMP, and destination interface for this session is ge-0/0/1.0).
- **Session ID 130000051**—Number that identifies the Layer 3 session. Use this ID to get more information about the Layer 3 session such as policy name or number of packets in and out.
- **default-policy-02/2**—Default policy name that permitted the Layer 3 traffic.
- **In**—Incoming flow (source and destination Layer 3 IP addresses with their respective source and destination port numbers, session is ICMP, and the source interface for this session is ge-0/0/2.0).
- **Out**—Reverse flow (source and destination Layer 3 IP addresses with their respective source and destination port numbers, session is ICMP, and destination interface for this session is ge-0/0/3.0).

**Related Documentation**

- Understanding Mixed Mode (Transparent and Route Mode) on Security Devices on page 11
- Understanding Secure Wire on Security Devices on page 75

PART 2

# Configuring Layer 2 Transparent Mode

CHAPTER 3

# Configuring Transparent Mode



## Understanding Layer 2 Transparent Mode on SRX Devices

**Supported Platforms** SRX Series, vSRX

For SRX Series devices, transparent mode provides full security services for Layer 2 switching capabilities. On these SRX Series devices, you can configure one or more VLANs to perform Layer 2 switching. A VLAN is a set of logical interfaces that share the same flooding or broadcast characteristics. Like a virtual LAN (VLAN), a VLAN spans one or more ports of multiple devices. Thus, the SRX Series device can function as a Layer 2 switch with multiple VLANs that participate in the same Layer 2 network.

In transparent mode, the SRX Series device filters packets that traverse the device without modifying any of the source or destination information in the IP packet headers. Transparent mode is useful for protecting servers that mainly receive traffic from untrusted sources because there is no need to reconfigure the IP settings of routers or protected servers.

In transparent mode, all physical ports on the device are assigned to Layer 2 interfaces. Do not route Layer 3 traffic through the device. Layer 2 zones can be configured to host Layer 2 interfaces, and security policies can be defined between Layer 2 zones. When packets travel between Layer 2 zones, security policies can be enforced on these packets.

Table 6 on page 26 lists the security features that are supported and are not supported in transparent mode for Layer 2 switching.

Table 6: Security Features Supported in Transparent Mode

| Mode Type | Supported | Not Supported |
|---|---|---|
| Transparent mode | • Application Layer Gateways (ALGs)<br>• Firewall User Authentication (FWAUTH)<br>• Intrusion Detection and Prevention (IDP)<br>• Screen<br>• AppSecure<br>• Unified Threat Management (UTM) | • Network Address Translation (NAT)<br>• VPN |

NOTE:

- On all SRX Series devices, transparent mode is not supported on mPIMs.
- On SRX300, SRX320, SRX340, SRX345, and SRX550M devices, the DHCP server propagation is not supported in Layer 2 transparent mode.

## Layer 2 Switching Exceptions on SRX Series Devices

The switching functions on the SRX Series devices are similar to the switching features on Juniper Networks MX Series routers. However, the following Layer 2 networking features on MX Series routers are not supported on SRX Series devices:

- Layer 2 control protocols—These protocols are used on MX Series routers for Rapid Spanning Tree Protocol (RSTP) or Multiple Spanning Tree Protocol (MSTP) in customer edge interfaces of a VPLS routing instance.
- Virtual switch routing instance—The virtual switching routing instance is used on MX Series routers to group one or more VLANs.
- Virtual private LAN services (VPLS) routing instance—The VPLS routing instance is used on MX Series routers for point-to-multipoint LAN implementations between a set of sites in a VPN.

In addition, the SRX Series devices do not support the following Layer 2 features:

- Spanning Tree Protocol (STP), RSTP, or MSTP—It is the user's responsibility to ensure that no flooding loops exist in the network topology.
- Internet Group Management Protocol (IGMP) snooping—Host-to-router signaling protocol for IPv4 used to report their multicast group memberships to neighboring routers and determine whether group members are present during IP multicasting.

- Double-tagged VLANs or IEEE 802.1Q VLAN identifiers encapsulated within 802.1Q packets (also called “Q in Q” VLAN tagging)—Only untagged or single-tagged VLAN identifiers are supported on SRX Series devices.
- Nonqualified VLAN learning, where only the MAC address is used for learning within the VLAN—VLAN learning on SRX Series devices is qualified; that is, both the VLAN identifier and MAC address are used.

Also, on SRX100, SRX110, SRX210, SRX220, SRX240, SRX300, SRX320, SRX340, SRX345, SRX550, or SRX650 devices, some features are not supported. (Platform support depends on the Junos OS release in your installation.) The following features are not supported for Layer 2 transparent mode on the mentioned devices:

- G-ARP on the Layer 2 interface
- IP address monitoring on any interface
- Transit traffic through IRB
- IRB interface in a routing instance
- IRB interface handling of Layer 3 traffic

NOTE: **The IRB interface is a pseudointerface and does not belong to the reth interface and redundancy group.**

## Layer 2 Transparent Mode on the SRX5000 Line Module Port Concentrator

The SRX5000 line Module Port Concentrator (SRX5K-MPC) supports Layer 2 transparent mode and processes the traffic when the SRX Series device is configured in Layer 2 transparent mode.

When the SRX5K-MPC is operating in Layer 2 mode, you can configure all interfaces on the SRX5K-MPC as Layer 2 switching ports to support Layer 2 traffic.

The security processing unit (SPU) supports all security services for Layer 2 switching functions, and the MPC delivers the ingress packets to the SPU and forwards the egress packets that are encapsulated by the SPU to the outgoing interfaces.

When the SRX Series device is configured in Layer 2 transparent mode, you can enable the interfaces on the MPC to work in Layer 2 mode by defining one or more logical units on a physical interface with the family address type as **Ethernet switching**. Later you can proceed with configuring Layer 2 security zones and configuring security policies in transparent mode. Once this is done, next-hop topologies are set up to process ingress and egress packets.

**Related Documentation**

- Understanding VLANs on Security Devices on page 29
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Understanding Layer 2 Interfaces on Security Devices on page 5
- Understanding Layer 2 Security Zones on page 37

- Understanding Security Policies in Transparent Mode on page 39

## Understanding Transparent Mode Conditions on Security Devices

**Supported Platforms** SRX Series, vSRX

There is no command to define or enable transparent mode on the device. The device operates in transparent mode when there are interfaces defined as Layer 2 interfaces. The device operates in route mode (the default mode) if there are no physical interfaces configured as Layer 2 interfaces.

NOTE: Starting in Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, mixed mode is the default mode, and you can configure an SRX Series device using both transparent mode (Layer 2) and route mode (Layer 3) simultaneously, with no reboot required.

You can configure the **fxp0** out-of-band management interface on the SRX Series device as a Layer 3 interface, even if Layer 2 interfaces are defined on the device. With the exception of the **fxp0** interface, you can define Layer 2 and Layer 3 interfaces on the device's network ports.

NOTE: There is no fxp0 out-of-band management interface on the SRX100. SRX210, SRX220. SRX240. SRX300, SRX320, SRX340, SRX345 or SRX650 devices. (Platform support depends on the Junos OS release in your installation.)

**Release History Table**

| Release | Description |
|---|---|
| 12.3X48-D10 | Starting in Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, mixed mode is the default mode, and you can configure an SRX Series device using both transparent mode (Layer 2) and route mode (Layer 3) simultaneously, with no reboot required. |

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Example: Configuring Layer 2 Logical Interfaces on Security Devices on page 6
- Understanding Layer 2 Interfaces on Security Devices on page 5
- Understanding Mixed Mode (Transparent and Route Mode) on Security Devices on page 11

CHAPTER 4

# Configuring VLANs in Transparent Mode



## Understanding VLANs on Security Devices

**Supported Platforms** SRX Series, vSRX

The packets that are forwarded within a VLAN are determined by the VLAN ID of the packets and the VLAN ID of the VLAN. Only the packets with VLAN IDs that match the VLAN ID configured for a VLAN are forwarded within the VLAN.

When configuring VLANs, you can specify either a single VLAN ID or a list of specific VLAN IDs. If you specify a list of VLAN IDs, a VLAN is created for each VLAN ID in the list. Certain VLAN properties, such as the integrated routing and bridging interface (IRB), are not configurable if VLANs are created in this manner.

Each Layer 2 logical interface configured on the device is implicitly assigned to a VLAN based on the VLAN ID of the packets accepted by the interface. You do not need to explicitly define the logical interfaces when configuring a VLAN.

You can configure one or more static MAC addresses for a logical interface in a VLAN; this is only applicable if you specified a single VLAN ID when creating the VLAN.

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.**

You can configure the following properties that apply to all VLANs on the SRX Series device:

- Layer 2 address learning—Layer 2 address learning is enabled by default. A VLAN learns unicast media access control (MAC) addresses to avoid flooding packets to all interfaces in the VLAN. Each VLAN creates a source MAC entry in its forwarding tables

for each source MAC address learned from packets received on interfaces that belong to the VLAN. When you disable MAC learning, source MAC addresses are not dynamically learned, and any packets sent to these source addresses are flooded into a VLAN.

- Maximum number of MAC addresses learned from all logical interfaces on the SRX Series device—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 instead. The default limits of MAC addresses for the SRX Series devices are shown in Table 7 on page 30 and Table 8 on page 30. (Platform support depends on the Junos OS release in your installation.)

### Table 7: MAC Addresses Default Limits for Junos OS Release 15.1X49-D30 and Earlier

| SRX Series Devices | Default Limit for MAC Addresses |
|---|---|
| SRX100<br>SRX210 | 1024 |
| SRX220 | 2048 |
| SRX240 | 4096 |
| SRX650 | 16,384 |
| SRX3400<br>SRX3600<br>SRX5600<br>SRX5800 | 131,071 |

Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, default limits for MAC addresses are more uniform.

### Table 8: MAC Addresses Default Limits for Junos OS Release Starting in Junos OS 15.1X49-D40

| SRX Series Devices | Default Limit for MAC Addresses |
|---|---|
| SRX300<br>SRX320<br>SRX340<br>SRX345 | 16,383 |
| SRX1500 | 24,575 |
| SRX4100 | 65536 |

**Table 8: MAC Addresses Default Limits for Junos OS Release Starting in Junos OS 15.1X49-D40 *(continued)***

| SRX Series Devices | Default Limit for MAC Addresses |
|---|---|
| SRX4200 | 65536 |
| SRX4600 | 65536 |
| SRX5600<br>SRX5800 | 131,071 |

- Timeout interval for MAC table entries. By default, the timeout interval for MAC table entries is 300 seconds. The minimum you can configure is 10 seconds and the maximum is 64,000 seconds. The timeout interval applies only to dynamically learned MAC addresses. This value does not apply to configured static MAC addresses, which never time out.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D40 | Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, default limits for MAC addresses are more uniform. |

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Example: Configuring VLANs on Security Devices on page 31
- Understanding Integrated Routing and Bridging Interfaces on a Security Device on page 7
- Understanding Layer 2 Interfaces on Security Devices on page 5
- Understanding Layer 2 Forwarding Tables on Security Devices on page 45

## Example: Configuring VLANs on Security Devices

**Supported Platforms** SRX Series, vSRX

This example shows how to configure VLANs.

NOTE: Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, new terminology and CLI keywords are used for switching functions. If your installation uses a Junos OS release preceding 15.1X49-D10, see "Enhanced Layer 2 CLI Configuration Statement and Command Changes for Security Devices" on page 34 to determine how you must modify configuration tasks for implementation in earlier Junos OS environments.



## Requirements

Before you begin, determine the properties you want to configure for the VLAN. See "Understanding VLANs on Security Devices" on page 29.

## Overview

In this example, you configure VLAN vlan-a for VLANs 1 and 10, and VLAN vlan-b for VLAN 2. You then limit the number of MAC addresses learned on all logical interfaces on the device to 64,000. When this limit is reached, incoming packets with a new source MAC address will be dropped.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans vlan-a vlan members 1-10
set vlans vlan-b vlan-id 2
set protocols l2-learning global-mac-limit 64000 packet-action drop
```

**Step-by-Step Procedure**

To configure VLANs:

1. Configure the domain type and VLANs.

   ```
   [edit]
   user@host# set vlans vlan-a vlan members 1-10
   user@host# set vlans vlan-b vlan-id 2
   ```

2. Limit the number of MAC addresses.

   ```
   [edit]
   user@host# set protocols l2-learning global-mac-limit 64000 packet-action drop
   ```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

## Verification

To verify the configuration is working properly, enter the **show vlans** and **show protocols l2-learning** commands.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D10 | Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, new terminology and CLI keywords are used for switching functions. |

**Related Documentation**

- Understanding Integrated Routing and Bridging Interfaces on a Security Device on page 7
- Understanding Layer 2 Interfaces on Security Devices on page 5
- Understanding Layer 2 Forwarding Tables on Security Devices on page 45
- Understanding VLANs on Security Devices on page 29

# Example: Configuring VLAN Retagging for Layer 2 Transparent Mode on a Security Device

**Supported Platforms** SRX Series, vSRX

This example shows how to configure VLAN retagging on a Layer 2 trunk interface to selectively screen incoming packets and redirect them to a security device without affecting other VLAN traffic.



## Requirements

Before you begin, determine the mapping you want to include for the VLAN retagging. See "Understanding VLAN Retagging on Security Devices" on page 120.

## Overview

In this example, you create a Layer 2 trunk interface called ge-3/0/0 and configure it to receive packets with VLAN identifiers 1 through 10. Packets that arrive on the interface with VLAN identifier 11 are retagged with VLAN identifier 2. Before exiting the trunk

interface, VLAN identifier 2 in the retagged packets is replaced with VLAN identifier 11. All VLAN identifiers in the retagged packets change back when you exit the trunk interface.

## Configuration

**Step-by-Step Procedure**

To configure VLAN retagging on a Layer 2 trunk interface:

1. Create a Layer 2 trunk interface.

   ```
   [edit]
   user@host#set interfaces ge-3/0/0 unit 0 family ethernet-switching interface-mode
     trunk vlan members 1–10
   ```

2. Configure VLAN retagging.

   ```
   [edit]
   user@host#set interfaces ge-3/0/0 unit 0 family ethernet-switching vlan-rewrite
     translate 11 2
   ```

3. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

## Verification

To verify the configuration is working properly, enter the **show interfaces ge-3/0/0** command.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Example: Configuring Layer 2 Logical Interfaces on Security Devices on page 6

# Enhanced Layer 2 CLI Configuration Statement and Command Changes for Security Devices

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, some Layer 2 CLI configuration statements are enhanced, and some commands are changed. Table 9 on page 35 and Table 10 on page 35 provide lists of existing commands that have been moved to new hierarchies or changed on SRX Series devices as part of this CLI enhancement effort. The tables are provided as a high-level reference only. For detailed information about these commands, see CLI Explorer.

Table 9: Enhanced Layer 2 Configuration Statement Changes

| Original Hierarchy | Changed Hierarchy | Hierarchy Level | Change Description |
|---|---|---|---|
| **bridge-domains**<br>  ***bridge-domain--name*** **{**<br>  ...<br>  }<br>} | **vlans *vlans-name* {**<br>  ...<br>  }<br>} | [edit] | Hierarchy renamed. |
| bridge-domains<br>  *bridge-domain--name* {<br>  **vlan-id-list [*vlan-id*]** ;<br>} | vlans *vlans-name* {<br>  **vlan members [*vlan-id*]** ;<br>} | [edit vlans *vlans-name*] | Statement renamed. |
| **bridge-options** {<br>  interface *interface-name* {<br>  encapsulation-type;<br>  ~~ignore-encapsulation-mismatch~~;<br>  pseudowire-status-tlv;<br>  static-mac *mac-address* {<br>  vlan-id *vlan-id*;<br>  }<br>  }<br>  mac-table-aging-time<br>  *seconds*;<br>  mac-table-size {<br>  *number*;<br>  packet-action drop;<br>  }<br>} | **switch-options** {<br>  interface *interface-name* {<br>  encapsulation-type;<br>  ~~ignore-encapsulation-mismatch~~;<br>  pseudowire-status-tlv;<br>  static-mac *mac-address* {<br>  vlan-id *vlan-id*;<br>  }<br>  }<br>  mac-table-aging-time<br>  *seconds*;<br>  mac-table-size {<br>  *number*;<br>  packet-action drop;<br>  }<br>} | [edit vlans *vlans-name*] | Statement renamed. |
| **bridge** {<br>  block-non-ip-all;<br>  bpdu-vlan-flooding;<br>  bypass-non-ip-unicast;<br>  no-packet-flooding {<br>  no-trace-route;<br>  }<br>} | **ethernet-switching** {<br>  block-non-ip-all;<br>  bpdu-vlan-flooding;<br>  bypass-non-ip-unicast;<br>  no-packet-flooding {<br>  no-trace-route;<br>  }<br>} | [edit security flow] | Statement renamed. |
| family {<br>  **bridge** {<br>  bridge-domain-type (svlan\|<br>  bvlan);<br>  ... | family {<br>  **ethernet-switching** {<br>  ... | [edit interfaces *interface-name* ]<br>unit *unit-number* | Hierarchy renamed. |
| ...<br>**routing-interface** irb.0;<br>... | ...<br>**l3-interface** irb.0;<br>... | [edit vlans *vlans-name*] | Statement renamed. |

Table 10: Enhanced Layer 2 Operational Command Changes

| Original Operational Command | Modified Operational Command |
|---|---|
| clear bridge mac-table | clear ethernet-switching table |
| clear bridge mac-table persistent-learning | clear ethernet-switching table persistent-learning |

Table 10: Enhanced Layer 2 Operational Command Changes *(continued)*

| Original Operational Command | Modified Operational Command |
|---|---|
| show bridge domain | show vlans |
| show bridge mac-table | show ethernet-switching table |
| show l2-learning interface | show ethernet-switching interface |

NOTE: There is no fxp0 out-of-band management interface on the SRX300, SRX320, and SRX500HM devices. (Platform support depends on the Junos OS release in your installation.)

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D10 | Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, some Layer 2 CLI configuration statements are enhanced, and some commands are changed. |

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Switching Modes on Security Devices on page 103

CHAPTER 5

# Configuring Security Zones and Security Policies



## Understanding Layer 2 Security Zones

**Supported Platforms** SRX Series, vSRX

A Layer 2 security zone is a zone that hosts Layer 2 interfaces. A security zone can be either a Layer 2 or Layer 3 zone; it can host either all Layer 2 interfaces or all Layer 3 interfaces, but it cannot contain a mix of Layer 2 and Layer 3 interfaces.

The security zone type—Layer 2 or Layer 3—is implicitly set from the first interface configured for the security zone. Subsequent interfaces configured for the same security zone must be the same type as the first interface.

NOTE: **You cannot configure a device with both Layer 2 and Layer 3 security zones.**

You can configure the following properties for Layer 2 security zones:

- Interfaces—List of interfaces in the zone.
- Policies—Active security policies that enforce rules for the transit traffic, in terms of what traffic can pass through the firewall, and the actions that need to take place on the traffic as it passes through the firewall.
- Screens—A Juniper Networks stateful firewall secures a network by inspecting, and then allowing or denying, all connection attempts that require passage from one security zone to another. For every security zone, and the MGT zone, you can enable a set of predefined screen options that detect and block various kinds of traffic that the device determines as potentially harmful.

NOTE: You can configure the same screen options for a Layer 2 security zone as for a Layer 3 security zone.

- Address books—IP addresses and address sets that make up an address book to identify its members so that you can apply policies to them.
- TCP-RST—When this feature is enabled, the system sends a TCP segment with the reset flag set when traffic arrives that does not match an existing session and does not have the synchronize flag set.

In addition, you can configure a Layer 2 zone for host-inbound traffic. This allows you to specify the kinds of traffic that can reach the device from systems that are directly connected to the interfaces in the zone. You must specify all expected host-inbound traffic because inbound traffic from devices directly connected to the device's interfaces is dropped by default.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Layer 2 Interfaces on Security Devices on page 5
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Example: Configuring Layer 2 Security Zones on page 38
- Example: Configuring Layer 2 Logical Interfaces on Security Devices on page 6

## Example: Configuring Layer 2 Security Zones

**Supported Platforms** SRX Series, vSRX

This example shows how to configure Layer 2 security zones.



### Requirements

Before you begin, determine the properties you want to configure for the Layer 2 security zone. See "Understanding Layer 2 Security Zones" on page 37.

### Overview

In this example, you configure security zone l2-zone1 to include a Layer 2 logical interface called ge-3/0/0.0 and security zone l2-zone2 to include a Layer 2 logical interface called ge-3/0/1.0. Then you configure l2-zone2 to allow all supported application services (such as SSH, Telnet, and SNMP) as host-inbound traffic.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security-zone l2-zone1 interfaces ge-3/0/0.0
set security-zone l2-zone2 interfaces ge-3/0/1.0
set security-zone l2–zone2 host-inbound-traffic system-services all
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Layer 2 security zones:

1. Create a Layer 2 security zone and assign interfaces to it.

   ```
   [edit security zones]
   user@host# set security-zone l2-zone1 interfaces ge-3/0/0.0
   user@host# set security-zone l2-zone2 interfaces ge-3/0/1.0
   ```

2. Configure one of the Layer 2 security zones.

   ```
   [edit security zones]
   user@host# set security-zone l2–zone2 host-inbound-traffic system-services all
   ```

3. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

## Verification

To verify the configuration is working properly, enter the **show security zones** command.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Example: Configuring Security Policies in Transparent Mode on page 41
- Example: Configuring Layer 2 Logical Interfaces on Security Devices on page 6

# Understanding Security Policies in Transparent Mode

**Supported Platforms** SRX Series, vSRX

In transparent mode, security policies can be configured only between Layer 2 zones. When packets are forwarded through the VLAN, the security policies are applied between

security zones. A security policy for transparent mode is similar to a policy configured for Layer 3 zones, with the following exceptions:

- NAT is not supported.
- IPsec VPN is not supported.
- Application ANY is used.

Layer 2 forwarding does not permit any interzone traffic unless there is a policy explicitly configured on the device. By default, Layer 2 forwarding performs the following actions:

- Allows or denies traffic specified by the configured policy.
- Allows Address Resolution Protocol (ARP) and Layer 2 non-IP multicast and broadcast traffic.
- Continues to block all non-IP and non-ARP unicast traffic.

This default behavior can be changed for Ethernet switching packet flow by using either J-Web or the CLI configuration editor:

- Configure the **block-non-ip-all** option to block all Layer 2 non-IP and non-ARP traffic, including multicast and broadcast traffic.
- Configure the **bypass-non-ip-unicast** option to allow all Layer 2 non-IP traffic to pass through the device.

NOTE: **You cannot configure both options at the same time.**

Starting in Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, you can create a separate security zone in mixed mode (the default mode) for Layer 2 and Layer 3 interfaces. However, there is no routing among IRB interfaces and between IRB interfaces and Layer 3 interfaces. Hence, you cannot configure security policies between Layer 2 and Layer 3 zones. You can only configure security policies between the Layer 2 zones or between Layer 3 zones.

**Release History Table**

| Release | Description |
|---|---|
| 12.3X48-D10 | Starting in Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, you can create a separate security zone in mixed mode (the default mode) for Layer 2 and Layer 3 interfaces. |

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Example: Configuring Security Policies in Transparent Mode on page 41
- Example: Configuring Layer 2 Security Zones on page 38

- Understanding Mixed Mode (Transparent and Route Mode) on Security Devices on page 11

## Example: Configuring Security Policies in Transparent Mode

**Supported Platforms** SRX Series, vSRX

This example shows how to configure security policies in transparent mode between Layer 2 zones.



### Requirements

Before you begin, determine the policy behavior you want to include in the Layer 2 security zone. See "Understanding Security Policies in Transparent Mode" on page 39.

### Overview

In this example, you configure a security policy to allow HTTP traffic from the 192.0.2.0/24 subnetwork in the l2–zone1 security zone to the server at 192.0.2.1/24 in the l2–zone2 security zone.

### 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security policies from-zone l2-zone1 to-zone l2-zone2 policy p1 match source-address
  192.0.2.0/24
set security policies from-zone l2-zone1 to-zone l2-zone2 policy p1 match
  destination-address 192.0.2.1/24
set security policies from-zone l2-zone1 to-zone l2-zone2 policy p1 match application http
set security policies from-zone l2-zone1 to-zone l2-zone2 policy p1 then permit
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure security policies in transparent mode:

1. Create policies and assign addresses to the interfaces for the zones.

```
[edit security policies]
user@host# set from-zone l2-zone1 to-zone l2-zone2 policy p1 match source-address
  192.0.2.0/24
```

```
user@host# set from-zone l2-zone1 to-zone l2-zone2 policy p1 match
  destination-address 192.0.2.1/24
```

2. Set policies for the application.

```
[edit security policies]
user@host# set from-zone l2-zone1 to-zone l2-zone2 policy p1 match application
  http
user@host# set from-zone l2-zone1 to-zone l2-zone2 policy p1 then permit
```

**Results** From configuration mode, confirm your configuration by entering the **show security policies** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host> show security policies
  from-zone l2-zone1 to-zone l2-zone2
{
  policy p1 {
    match {
    source-address 192.0.2.0/24;
    destination-address 192.0.2.1/24;
    application junos-http;
  }
  then {
    permit;
  }
}
}
```

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

## Verification

### Verifying Layer 2 Security Policies

**Purpose** Verify that the Layer 2 security policies are configured properly.

**Action** From configuration mode, enter the **show security policies** command.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Example: Configuring Layer 2 Security Zones on page 38

# Understanding Firewall User Authentication in Transparent Mode

**Supported Platforms** SRX Series

A firewall user is a network user who must provide a username and password for authentication when initiating a connection across the firewall. Firewall user authentication enables administrators to restrict and permit users accessing protected resources behind a firewall based on their source IP address and other credentials. Junos OS supports the following types of firewall user authentication for transparent mode on the SRX Series device:

- Pass-through authentication—A host or a user from one zone tries to access resources on another zone. You must use an FTP, Telnet, or HTTP client to access the IP address of the protected resource and be authenticated by the firewall. The device uses FTP, Telnet, or HTTP to collect username and password information, and subsequent traffic from the user or host is allowed or denied based on the result of this authentication.
- Web authentication—Users try to connect, by using HTTP, to an IP address on the IRB interface that is enabled for Web authentication. You are prompted for the username and password that are verified by the device. Subsequent traffic from the user or host to the protected resource is allowed or denied based on the result of this authentication.

**Related Documentation**

- *Authentication and Integrated User Firewalls Feature Guide for Security Devices*
- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Integrated Routing and Bridging Interfaces on a Security Device on page 7
- Example: Configuring an IRB Interface on a Security Device on page 8

CHAPTER 6

# Configuring Layer 2 Forwarding Tables



## Understanding Layer 2 Forwarding Tables on Security Devices

**Supported Platforms** SRX Series, vSRX

The SRX Series device maintains forwarding tables that contain MAC addresses and associated interfaces for each Layer 2 VLAN. When a packet arrives with a new source MAC address in its frame header, the device adds the MAC address to its forwarding table and tracks the interface at which the packet arrived. The table also contains the corresponding interface through which the device can forward traffic for a particular MAC address.

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 VLAN 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. Packet flooding is performed at two levels: packets are flooded to different zones as permitted by configured Layer 2 security policies, and packets are also flooded to different interfaces with the same VLAN identifier within the same zone. The device learns the forwarding interface for the MAC address when a reply with that MAC address arrives at one of its interfaces.

You can specify that the SRX Series device use ARP queries and traceroute requests (which are ICMP echo requests with the time-to-live values set to 1) instead of packet flooding to locate an unknown destination MAC address. This method is considered more secure than packet flooding because the device floods ARP queries and traceroute packets—not the initial packet—on all interfaces. When ARP or traceroute flooding is used, the original packet is dropped. The device broadcasts an ARP or ICMP query to all other devices on the same subnetwork, requesting the device at the specified destination IP address to send back a reply. Only the device with the specified IP address replies, which provides the requestor with the MAC address of the responder.

ARP allows the device to discover the destination MAC address for a unicast packet if the destination IP address is in the same subnetwork as the ingress IP address. (The ingress IP address refers to the IP address of the last device to send the packet to the

device. The device might be the source that sent the packet or a router forwarding the packet.) Traceroute allows the device to discover the destination MAC address even if the destination IP address belongs to a device in a subnetwork beyond that of the ingress IP address.

When you enable ARP queries to locate an unknown destination MAC address, traceroute requests are also enabled. You can also optionally specify that traceroute requests not be used; however, the device can then discover destination MAC addresses for unicast packets only if the destination IP address is in the same subnetwork as the ingress IP address.

Whether you enable ARP queries and traceroute requests or ARP-only queries to locate unknown destination MAC addresses, the SRX Series device performs the following series of actions:

1. The device notes the destination MAC address in the initial packet. The device adds the source MAC address and its corresponding interface to its forwarding table, if they are not already there.

2. The device drops the initial packet.

3. The device generates an ARP query packet and optionally a traceroute packet and floods those packets out all interfaces except the interface on which the initial packet arrived.

   ARP packets are sent out with the following field values:

   - Source IP address set to the IP address of the IRB
   - Destination IP address set to the destination IP address of the original packet
   - Source MAC address set to the MAC address of the IRB
   - Destination MAC address set to the broadcast MAC address (all **0xf**)

   Traceroute (ICMP echo request or ping) packets are sent out with the following field values:

   - Source IP address set to the IP address of the original packet
   - Destination IP address set to the destination IP address of the original packet
   - Source MAC address set to the source MAC address of the original packet
   - Destination MAC address set to the destination MAC address of the original packet
   - Time-to-live (TTL) set to **1**

4. Combining the destination MAC address from the initial packet with the interface leading to that MAC address, the device adds a new entry to its forwarding table.

5. The device forwards all subsequent packets it receives for the destination MAC address out the correct interface to the destination.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Integrated Routing and Bridging Interfaces on a Security Device on page 7
- Example: Configuring an IRB Interface on a Security Device on page 8
- Example: Configuring the Default Learning for Unknown MAC Addresses on page 47

## Example: Configuring the Default Learning for Unknown MAC Addresses

**Supported Platforms** SRX Series

This example shows how to configure the device to use only ARP requests to learn the outgoing interfaces for unknown destination MAC addresses.



### Requirements

Before you begin, determine the MAC addresses and associated interfaces of the forwarding table. See "Understanding Layer 2 Forwarding Tables on Security Devices" on page 45.

### Overview

In this example, you configure the device to use only ARP queries without traceroute requests.

### 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security flow ethernet-switching no-packet-flooding no-trace-route
```

**Step-by-Step Procedure**

To configure the device to use only ARP requests to learn unknown destination MAC addresses:

1. Enable the device.

   ```
   [edit]
   user@host# set security flow ethernet-switching no-packet-flooding no-trace-route
   ```

2. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   ```

```
user@host# commit
```

## Verification

To verify the configuration is working properly, enter the **show security flow** command.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Integrated Routing and Bridging Interfaces on a Security Device on page 7
- Example: Configuring an IRB Interface on a Security Device on page 8

CHAPTER 7

# Configuring Layer 2 Transparent Mode Chassis Clusters



## Understanding Layer 2 Transparent Mode Chassis Clusters on Security Devices

**Supported Platforms** SRX Series, vSRX

A pair of SRX Series devices in Layer 2 transparent mode can be connected in a chassis cluster to provide network node redundancy. When configured in a chassis cluster, one node acts as the primary device and the other as the secondary device, ensuring stateful failover of processes and services in the event of system or hardware failure. If the primary device fails, the secondary device takes over processing of traffic.

NOTE: **If the primary device fails in a Layer 2 transparent mode chassis cluster, the physical ports in the failed device become inactive (go down) for a few seconds before they become active (come up) again.**

To form a chassis cluster, a pair of the same kind of supported SRX Series devices combines to act as a single system that enforces the same overall security.

Devices in Layer 2 transparent mode can be deployed in active/backup and active/active chassis cluster configurations.

The following chassis cluster features are not supported for devices in Layer 2 transparent mode:

- Gratuitous ARP—The newly elected master in a redundancy group cannot send gratuitous ARP requests to notify network devices of a change in mastership on the redundant Ethernet interface links.
- IP address monitoring—Failure of an upstream device cannot be detected.

A redundancy group is a construct that includes a collection of objects on both nodes. A redundancy group is primary on one node and backup on the other. When a redundancy group is primary on a node, its objects on that node are active. When a redundancy group fails over, all its objects fail over together.

You can create one or more redundancy groups numbered 1 through 128 for an active/active chassis cluster configuration. Each redundancy group contains one or more redundant Ethernet interfaces. A redundant Ethernet interface is a pseudointerface that contains physical interfaces from each node of the cluster. The physical interfaces in a redundant Ethernet interface must be the same kind—either Fast Ethernet or Gigabit Ethernet. If a redundancy group is active on node 0, then the child links of all associated redundant Ethernet interfaces on node 0 are active. If the redundancy group fails over to the node 1, then the child links of all redundant Ethernet interfaces on node 1 become active.

NOTE: In the active/active chassis cluster configuration, the maximum number of redundancy groups is equal to the number of redundant Ethernet interfaces that you configure. In the active/backup chassis cluster configuration, the maximum number of redundancy groups supported is two.

Configuring redundant Ethernet interfaces on a device in Layer 2 transparent mode is similar to configuring redundant Ethernet interfaces on a device in Layer 3 route mode, with the following difference: the redundant Ethernet interface on a device in Layer 2 transparent mode is configured as a Layer 2 logical interface.

The redundant Ethernet interface may be configured as either an access interface (with a single VLAN ID assigned to untagged packets received on the interface) or as a trunk interface (with a list of VLAN IDs accepted on the interface and, optionally, a native-vlan-id for untagged packets received on the interface). Physical interfaces (one from each node in the chassis cluster) are bound as child interfaces to the parent redundant Ethernet interface.

In Layer 2 transparent mode, MAC learning is based on the redundant Ethernet interface. The MAC table is synchronized across redundant Ethernet interfaces and Services Processing Units (SPUs) between the pair of chassis cluster devices.

The IRB interface is used only for management traffic, and it cannot be assigned to any redundant Ethernet interface or redundancy group.

All Junos OS screen options that are available for a single, nonclustered device are available for devices in Layer 2 transparent mode chassis clusters.

NOTE: Spanning Tree Protocols (STPs) are not supported for Layer 2 transparent mode. You must ensure that there are no loop connections in the deployment topology.

**Related Documentation**

- *Chassis Cluster Feature Guide for SRX Series Devices*
- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Layer 2 Interfaces on Security Devices on page 5
- Example: Configuring Layer 2 Logical Interfaces on Security Devices on page 6
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Example: Configuring Redundant Ethernet Interfaces for Layer 2 Transparent Mode Chassis Clusters on Security Devices on page 51
- Understanding Layer 2 Forwarding Tables on Security Devices on page 45

## Example: Configuring Redundant Ethernet Interfaces for Layer 2 Transparent Mode Chassis Clusters on Security Devices

**Supported Platforms** SRX Series, vSRX

This example shows how to configure a redundant Ethernet interface on a device as a Layer 2 logical interface for a Layer 2 transparent mode chassis cluster.



### Requirements

Before you begin, determine the devices you want to connect in a chassis cluster. See "Understanding Layer 2 Transparent Mode Chassis Clusters on Security Devices" on page 49.

### Overview

This example shows you how to configure the redundant Ethernet interface as a Layer 2 logical interface and how to bind the physical interfaces (one from each node in the chassis cluster) to the redundant Ethernet interface. In this example, you create redundant Ethernet interface reth0 for redundancy group 1 and configure reth0 as an access interface with the VLAN identifier 1. Then you assign physical interface ge-2/0/2 on a chassis cluster node to the redundant Ethernet interface reth0.

### 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set reth0 redundant-ether-options redundancy-group 1
set reth0 unit 0 family ethernet-switching interface-mode access vlan-id 1
set ge-2/0/2 gigether-options redundant-parent reth0
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a redundant Ethernet interface as a Layer 2 logical interface:

1. Configure the interfaces and redundancy group.

```
[edit interfaces]
user@host# set reth0 redundant-ether-options redundancy-group 1
user@host# set reth0 unit 0 family ethernet-switching interface-mode access
  vlan-id 1
```

2. Assign a physical interface on a chassis cluster node.

```
[edit interfaces]
user@host# set ge-2/0/2 gigether-options redundant-parent reth0
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

## Verification

To verify the configuration is working properly, enter the **show interfaces reth0** and **show interfaces ge-2/0/2** commands.

**Related Documentation**

- *Chassis Cluster Feature Guide for SRX Series Devices*
- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Understanding Layer 2 Transparent Mode Chassis Clusters on Security Devices on page 49
- Understanding Layer 2 Forwarding Tables on Security Devices on page 45

CHAPTER 8

# Configuring IP Spoofing in Layer 2 Transparent Mode



## Understanding IP Spoofing in Layer 2 Transparent Mode on Security Devices

**Supported Platforms** SRX Series, vSRX

In an IP spoofing attack, the attacker gains access to a restricted area of the network and inserts a false source address in the packet header to make the packet appear to come from a trusted source. IP spoofing is most frequently used in denial-of-service (DoS) attacks. When SRX Series devices are operating in transparent mode, the IP spoof-checking mechanism makes use of address book entries. Address books only exist on the Routing Engine. IP spoofing in Layer 2 transparent mode is performed on the Packet Forwarding Engine. Address book information cannot be obtained from the Routing Engine each time a packet is received by the Packet Forwarding Engine. Therefore, address books attached to the Layer 2 zones must be pushed to the Packet Forwarding Engine.

NOTE: **IP spoofing in Layer 2 transparent mode does not support DNS and wildcard addresses.**

When a packet is received by the Packet Forwarding Engine, the packet's source IP address is checked to determine if it is in the incoming zone's address-book. If the packet's source IP address is in the incoming zone's address book, then this IP address is allowed on the interface, and traffic is passed.

If the source IP address is not present in the incoming zone's address-book, but exists in other zones', then the IP address is considered a spoofed IP. Accordingly, actions such as drop and logging can be taken depending on the screen configuration (alarm-without-drop).

NOTE: **If the alarm-without-drop option is configured, the Layer 2 spoofing packet only triggers an alarm message, but the packet is not dropped.**

If a packet's source IP address is not present in the incoming zone's address book or other zones', then you cannot determine if the IP is spoofed or not. In such instances, the packet is passed.

Junos OS takes into account the following match conditions while it searches for source IP addresses in the address book:

- **Host-match**—The IP address match found in the address-book is an address without a prefix.
- **Prefix-match**—The IP address match found in the address-book is an address with a prefix.
- **Any-match**—The IP address match found in the address-book is "any", "any-IPv4", or "any-IPv6".
- **No-match**—No IP address match is found.

## Configuring IP Spoofing in Layer 2 Transparent Mode on Security Devices

**Supported Platforms** SRX Series, vSRX

You can configure the IP spoof-checking mechanism to determine whether or not an IP is being spoofed.

To configure IP spoofing in Layer 2 transparent mode:

1. Set the interface in Layer 2 transparent mode.

   ```
   [edit]
   user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching
   ```

2. (Optional) Set the zone in Layer 2 transparent mode.

   ```
   [edit]
   user@host# set security zones security-zone untrust interfaces ge-0/0/1.0
   ```

3. Configure the address book.

   ```
   [edit]
   user@host# set security address-book my-book address myadd1 10.1.1.0/24
   user@host# set security address-book my-book address myadd2 10.1.2.0/24
   ```

4. Apply the address book to the zone.

   ```
   [edit]
   user@host# set security address-book my-book attach zone untrust
   ```

5. Configure screen IP spoofing.

   ```
   [edit]
   user@host# set security screen ids-option my-screen ip spoofing
   ```

6. Apply the screen to the zone.

```
[edit]
user@host# set security zones security-zone untrust screen my-screen
```

7. (Optional) Configure the **alarm-without-drop** option.

```
[edit]
user@host# set security screen ids-option my-screen alarm-without-drop
```

NOTE: If the **alarm-without-drop** option is configured, the Layer 2 spoofing packet only triggers an alarm message, but the packet is not dropped.

CHAPTER 9

# Configuring Class of Service in Transparent Mode



## Class of Service Functions in Transparent Mode Overview

**Supported Platforms** SRX Series, vSRX

Devices operating in Layer 2 transparent mode support the following class-of-service (CoS) functions:

- IEEE 802.1p behavior aggregate (BA) classifiers to determine the forwarding treatment for packets entering the device

  NOTE: **Only IEEE 802.1p BA classifier types are supported on devices operating in transparent mode.**

- Rewrite rules to redefine IEEE 802.1 CoS values in outgoing packets

  NOTE: **Rewrite rules that redefine IP precedence CoS values and Differentiated Services Code Point (DSCP) CoS values are not supported on devices operating in transparent mode.**

- Shapers to apply rate limiting to an interface
- Schedulers that define the properties of an output queue

You configure BA classifiers and rewrite rules on transparent mode devices in the same way as on devices operating in Layer 3 mode. For transparent mode devices, however,

you apply BA classifiers and rewrite rules only to logical interfaces configured with the **family ethernet-switching** configuration statement.

**Related Documentation**

- *Class of Service Feature Guide for Security Devices*
- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Understanding BA Traffic Classification on Transparent Mode Security Devices on page 58
- Example: Configuring BA Classifiers on Transparent Mode Security Devices on page 59

## Understanding BA Traffic Classification on Transparent Mode Security Devices

**Supported Platforms** SRX Series, vSRX

A BA classifier checks the header information of an ingress packet. The resulting traffic classification consists of a forwarding class (FC) and packet loss priority (PLP). The FC and PLP associated with a packet specify the CoS behavior of a hop within the system. For example, a hop can place a packet into a priority queue according to its FC, and manage queues by checking the packet's PLP. Junos OS supports up to eight FCs and four PLPs.

NOTE: MPLS EXP bit-based traffic classification is not supported.

BA classification can be applied within one DiffServ domain. BA classification can also be applied between two domains, where each domain honors the CoS results generated by the other domain. Junos OS performs BA classification for a packet by examining its Layer 2 and Layer 3 CoS-related parameters. Those parameters include the following:

- Layer 2—IEEE 802.1p: User Priority
- Layer 3—IPv4 Precedence, IPv4 DSCP, IPv6 DSCP

On SRX Series devices in transparent mode, a BA classifier evaluates only Layer 2 parameters. On SRX Series devices in Layer 3 mode, a BA classifier can evaluate Layer 2 and Layer 3 parameters; in that case, classification resulting from Layer 3 parameters overrides that of Layer 2 parameters.

On SRX Series devices in transparent mode, you specify one of four PLP levels—high, medium-high, medium-low, or low—when configuring a BA classifier.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Class of Service Functions in Transparent Mode Overview on page 57
- Example: Configuring BA Classifiers on Transparent Mode Security Devices on page 59

## Example: Configuring BA Classifiers on Transparent Mode Security Devices

**Supported Platforms** SRX Series

This example shows how to configure BA classifiers on transparent mode devices to determine the forwarding treatment of packets entering the devices.



### Requirements

Before you begin, configure a Layer 2 logical interface. See "Example: Configuring Layer 2 Logical Interfaces on Security Devices" on page 6.

### Overview

In this example, you configure logical interface ge-0/0/4.0 as a trunk port that carries traffic for packets tagged with VLAN identifiers 200 through 390. You then configure forwarding classes and create BA classifier c1 for IEEE 802.1 traffic where incoming packets with IEEE 802.1p priority bits 110 are assigned to the forwarding class fc1 with a low loss priority. Finally, you apply the BA classifier c1 to interface ge-0/0/4.0.

### 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/4 vlan-tagging unit 0 family ethernet-switching interface-mode
    trunk vlan members 200-390
set class-of-service forwarding-classes queue 0 fc1
set class-of-service forwarding-classes queue 1 fc2
set class-of-service forwarding-classes queue 3 fc4
set class-of-service forwarding-classes queue 4 fc5
set class-of-service forwarding-classes queue 5 fc6
set class-of-service forwarding-classes queue 6 fc7
set class-of-service forwarding-classes queue 7 fc8
set class-of-service forwarding-classes queue 2 fc3
set class-of-service classifiers ieee-802.1 c1 forwarding-class fc1 loss-priority low
    code-point 110
set class-of-service interfaces ge-0/0/4 unit 0 classifiers ieee-802.1 c1
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure BA classifiers on transparent mode devices:

1. Configure the logical interface as a Layer 2 trunk port.

```
[edit]
user@host# set interfaces ge-0/0/4 vlan-tagging unit 0 family ethernet-switching
  interface-mode trunk vlan members 200-390
```

2. Configure the class of service.

```
[edit]
user@host# edit class-of-service
```

3. Configure the forwarding classes.

```
[edit class-of-service]
user@host# set forwarding-classes queue 0 fc1
user@host# set forwarding-classes queue 1 fc2
user@host# set forwarding-classes queue 3 fc4
user@host# set forwarding-classes queue 4 fc5
user@host# set forwarding-classes queue 5 fc6
user@host# set forwarding-classes queue 6 fc7
user@host# set forwarding-classes queue 7 fc8
user@host# set forwarding-classes queue 2 fc3
```

4. Configure a BA classifier.

```
[edit class-of-service]
user@host# set classifiers ieee-802.1 c1 forwarding-class fc1 loss-priority low
  code-points 110
```

5. Apply the BA classifier to the interface.

```
[edit class-of-service]
user@host# set interfaces ge-0/0/4 unit 0 classifiers ieee-802.1 c1
```

**Results**

From configuration mode, confirm your configuration by entering the **show interfaces ge-0/0/4** and **show class-of-service** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host> show interfaces ge-0/0/4
  vlan-tagging;
  unit 0 {
    family ethernet-switching {
    interface-mode trunk;
    vlan members 200-390;
  }
```

```
}
[edit]
  user@host> show class-of-service
  classifiers {
  ieee-802.1 c1 {
    forwarding-class fc1 {
    loss-priority low code-points 110;
    }
  }
  }
  forwarding-classes {
    queue 0 fc1;
    queue 1 fc2;
    queue 3 fc4;
    queue 4 fc5;
    queue 5 fc6;
    queue 6 fc7;
    queue 7 fc8;
    queue 2 fc3;
    }
    interfaces {
      ge-0/0/4 {
    unit 0 {
    classifiers {
      ieee-802.1 c1;
    }
    }
  }
}
```

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

## Verification

To confirm that the configuration is working properly, perform this task:

### Verifying BA Classifiers on Transparent Mode Devices

**Purpose** Verify that the BA classifier was configured on the transparent mode devices properly.

**Action** From configuration mode, enter the **show interfaces ge-0/0/4** and **show class-of-service** commands.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Class of Service Functions in Transparent Mode Overview on page 57
- Understanding BA Traffic Classification on Transparent Mode Security Devices on page 58

## Understanding Rewrite of Packet Headers on Transparent Mode Security Devices

**Supported Platforms** SRX Series, vSRX

Before a packet is transmitted from an interface, the CoS fields in the packet's header can be rewritten for the forwarding class (FC) and packet loss priority (PLP) of the packet. The rewriting function converts a packet's FC and PLP into corresponding CoS fields in the packet header. In Layer 2 transparent mode, the CoS fields are the IEEE 802.1p priority bits.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Example: Configuring Rewrite Rules on Transparent Mode Security Devices on page 62

## Example: Configuring Rewrite Rules on Transparent Mode Security Devices

**Supported Platforms** SRX Series, vSRX

This example shows how to configure rewrite rules on transparent mode devices to redefine IEEE 802.1 CoS values in outgoing packets.



### Requirements

Before you begin, configure a Layer 2 logical interface. See "Example: Configuring Layer 2 Logical Interfaces on Security Devices" on page 6.

### Overview

In this example, you configure logical interface ge-1/0/3.0 as a trunk port that carries traffic for packets tagged with VLAN identifiers 200 through 390. You then configure the forwarding classes and create rewrite rule rw1 for IEEE 802.1 traffic. For outgoing packets in the forwarding class fc1 with low loss priority, the IEEE 802.1p priority bits are rewritten as 011. Finally, you apply the rewrite rule rw1 to interface ge-1/0/3.0.

### 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-1/0/3 vlan-tagging unit 0 family ethernet-switching interface-mode
    trunk vlan members 200-390
```

```
set class-of-service forwarding-classes queue 0 fc1
set class-of-service forwarding-classes queue 1 fc2
set class-of-service forwarding-classes queue 3 fc4
set class-of-service forwarding-classes queue 4 fc5
set class-of-service forwarding-classes queue 5 fc6
set class-of-service forwarding-classes queue 6 fc7
set class-of-service forwarding-classes queue 7 fc8
set class-of-service forwarding-classes queue 2 fc3
set class-of-service rewrite-rules ieee-802.1 rw1 forwarding-class fc1 loss-priority low
  code-point 011
set class-of-service interfaces ge-1/0/3 unit 0 rewrite-rules ieee-802.1 rw1
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure rewrite rules on transparent mode devices:

1. Configure the logical interface as a Layer 2 trunk port.

```
[edit]
user@host# set interfaces ge-1/0/3 vlan-tagging unit 0 family ethernet-switching
  interface-mode trunk vlan members 200-390
```

2. Configure the class of service.

```
[edit]
user@host# edit class-of-service
```

3. Configure the forwarding classes.

```
[edit class-of-service]
user@host# set forwarding-classes queue 0 fc1
user@host# set forwarding-classes queue 1 fc2
user@host# set forwarding-classes queue 3 fc4
user@host# set forwarding-classes queue 4 fc5
user@host# set forwarding-classes queue 5 fc6
user@host# set forwarding-classes queue 6 fc7
user@host# set forwarding-classes queue 7 fc8
user@host# set forwarding-classes queue 2 fc3
```

4. Configure a rewrite rule.

```
[edit class-of-service]
user@host# set rewrite-rules ieee-802.1 rw1 forwarding-class fc1 loss-priority low
  code-point 011
```

5. Apply the rewrite rule to the interface.

```
[edit class-of-service]
user@host# set interfaces ge-1/0/3 unit 0 rewrite-rules ieee-802.1 rw1
```

**Results** From configuration mode, confirm your configuration by entering the **show interfaces ge-1/0/3** and **show class-of-service** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host> show interfaces ge-1/0/3
  vlan-tagging;
  unit 0 {
    family ethernet-switching {
    interface-mode trunk;
    vlan members 200-390;
  }
}
[edit]
  user@host> show class-of-service
  forwarding-classes {
  queue 0 fc1;
    queue 1 fc2;
    queue 3 fc4;
    queue 4 fc5;
    queue 5 fc6;
    queue 6 fc7;
    queue 7 fc8;
  queue 2 fc3;
  }
  interfaces {
    ge-1/0/3 {
    unit 0 {
    rewrite-rules {
      ieee-802.1 rw1;
      }
    }
  }
  }
  rewrite-rules {
    ieee-802.1 rw1 {
    forwarding-class fc1 {
    loss-priority low code-point 011;
    }
  }
}
```

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

## Verification

To confirm that the configuration is working properly, perform this task:

### Verifying Rewrite Rules on Transparent Mode Devices

**Purpose** Verify that the rewrite rule was configured on the transparent mode devices properly.

**Action** From configuration mode, enter the **show interfaces ge-1/0/3** and **show class-of-service** commands.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Understanding Transparent Mode Conditions on Security Devices on page 28
- Understanding Rewrite of Packet Headers on Transparent Mode Security Devices on page 62

CHAPTER 10

# Configuring IPv6 Flows



## Understanding IPv6 Flows in Transparent Mode on Security Devices

**Supported Platforms** SRX Series

In transparent mode, the SRX Series device filters packets that traverse the device without modifying any of the source or destination information in the packet MAC headers. Transparent mode is useful for protecting servers that mainly receive traffic from untrusted sources because there is no need to reconfigure the IP settings of routers or protected servers.

A device operates in transparent mode when all physical interfaces on the device are configured as Layer 2 interfaces. A physical interface is a Layer 2 interface if its logical interface is configured with the **ethernet-switching** option at the [**edit interfaces** ***interface-name*** **unit** ***unit-number*** **family**] hierarchy level. There is no command to define or enable transparent mode on the device. The device operates in transparent mode when there are interfaces defined as Layer 2 interfaces. The device operates in route mode (the default mode) if all physical interfaces are configured as Layer 3 interfaces.

By default, IPv6 flows are dropped on security devices. To enable processing by security features such as zones, screens, and firewall policies, you must enable flow-based forwarding for IPv6 traffic with the **mode flow-based** configuration option at the [**edit security forwarding-options family inet6**] hierarchy level. You must reboot the device when you change the mode.

In transparent mode, you can configure Layer 2 zones to host Layer 2 interfaces, and you can define security policies between Layer 2 zones. When packets travel between Layer 2 zones, security policies can be enforced on these packets. The following security features are supported for IPv6 traffic in transparent mode:

- Layer 2 security zones and security policies. See "Understanding Layer 2 Security Zones" on page 37 and "Understanding Security Policies in Transparent Mode" on page 39.
- Firewall user authentication. See "Understanding Firewall User Authentication in Transparent Mode" on page 42.

- Layer 2 transparent mode chassis clusters. See "Understanding Layer 2 Transparent Mode Chassis Clusters on Security Devices" on page 49.
- Class of service functions. See "Class of Service Functions in Transparent Mode Overview" on page 57.

The following security features are *not* supported for IPv6 flows in transparent mode:

- Logical systems
- IPv6 GTPv2
- J-Web interface
- NAT
- IPsec VPN
- With the exception of DNS, FTP, and TFTP ALGs, all other ALGs are not supported.

Configuring VLANs and Layer 2 logical interfaces for IPv6 flows is the same as configuring VLANs and Layer 2 logical interfaces for IPv4 flows. You can optionally configure an integrated routing and bridging (IRB) interface for management traffic in a VLAN. The IRB interface is the only Layer 3 interface allowed in transparent mode. The IRB interface on the SRX Series device does not support traffic forwarding or routing. The IRB interface can be configured with both IPv4 and IPv6 addresses. You can assign an IPv6 address for the IRB interface with the **address** configuration statement at the [**edit interfaces irb unit *number* family inet6**] hierarchy level. You can assign an IPv4 address for the IRB interface with the **address** configuration statement at the [**edit interfaces irb unit *number* family inet**] hierarchy level.

The Ethernet Switching functions on SRX Series devices are similar to the switching features on Juniper Networks MX Series routers. However, not all Layer 2 networking features supported on MX Series routers are supported on SRX Series devices. See "Understanding Layer 2 Transparent Mode on SRX Devices" on page 25.

The SRX Series device maintains forwarding tables that contain MAC addresses and associated interfaces for each Layer 2 VLAN. The IPv6 flow processing is similar to IPv4 flows. See "Understanding Layer 2 Forwarding Tables on Security Devices" on page 45.

**Related Documentation**

- Flow-Based Processing for IPv6 Traffic on Security Devices on page 68
- Example: Configuring Transparent Mode for IPv6 Flows on Security Devices on page 70

## Flow-Based Processing for IPv6 Traffic on Security Devices

**Supported Platforms** SRX Series, vSRX

Flow-based processing mode is required for security features such as zones, screens, and firewall policies to function. By default, the SRX Series device is enabled for flow-based forwarding for IPv6 traffic on all devices, apart from the SRX300 Series and SRX550M devices that are set to drop mode. Starting with Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, for the SRX1500 series, SRX4100, SRX4200, SRX5400,

SRX5600, SRX5800 and vSRX devices, you do *not* need to reboot the device when you are switching modes between flow mode, packet mode, and drop mode. For SRX300 Series and SRX550M devices, you *must* reboot the device when switching between flow mode, packet mode, and drop mode.

### SRX300 Series and the SRX550M Devices

When IPv6 is configured on SRX300 Series and the SRX550M devices, the default behavior is set to drop mode because of memory constraints. In this case, you must reboot the device after changing the processing mode from the drop mode default to flow-based processing mode or packet-based processing mode—that is, between modes on these devices.

NOTE: For drop mode processing, the traffic is dropped directly, it is not forwarded. It differs from packet-mode processing for which the traffic is handled but no security processes are applied.

To process IPv6 traffic on SRX300 Series and the SRX550M devices, you need to configure IPv6 addresses for the transit interfaces that receive and forward the traffic. For information about the inet6 protocol family and procedures for configuring IPv6 addresses for interfaces, see *Interfaces Feature Guide for Security Devices*.

### Configuring an SRX Series Device as a Border Router

When an SRX Series device of any type is enabled for flow-based processing or drop mode, to configure the device as a border router you must change the mode to packet-based processing for MPLS. In this case, to configure the SRX device to packet mode for MPLS, use the **set security forwarding-options family mpls mode packet-based** statement.

NOTE: As mentioned, for SRX300 Series and the SRX550M devices, whenever you change processing modes, you must reboot the device.

### Enabling Flow-Based Processing for IPv6 Traffic on SRX300 Series and SRX550M Devices

To enable flow-based forwarding for IPv6 traffic on SRX300 Series and the SRX550M devices, modify the mode at the [**edit security forwarding-options family inet6**] hierarchy level:

```
security {
  forwarding-options {
    family {
      inet6 {
        mode flow-based;
      }
    }
  }
}
```

To configure forwarding for IPv6 traffic on SRX300 Series or an SRX500M device:

1. Change the forwarding option mode for IPv6 to flow-based.

```
[edit]
user@host# security forwarding-options family inet6 mode flow-based
```

2. Review your configuration.

```
[edit]
user@host# show security forwarding-options
  family {
    inet6 {
      mode flow-based;
    }
  }
```

3. Commit the configuration.

```
[edit]
user@host# commit
```

4. Reboot the device.

NOTE: For SRX300 Series and SRX500M devices, the device discards IPv6 type 0 Routing Header (RH0) packets.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D70 | By default, the SRX Series device is enabled for flow-based forwarding for IPv6 traffic on all devices, apart from the SRX300 Series and SRX550M devices that are set to drop mode. Starting with Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, for the SRX1500 series, SRX4100, SRX4200, SRX5400, SRX5600, SRX5800 and vSRX devices, you do *not* need to reboot the device when you are switching modes between flow mode, packet mode, and drop mode. For SRX300 Series and SRX550M devices, you *must* reboot the device when switching between flow mode, packet mode, and drop mode. |

**Related Documentation**

- *Understanding IPv6 Address Space, Addressing, Address Format, and Address Types*
- *Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways*

## Example: Configuring Transparent Mode for IPv6 Flows on Security Devices

**Supported Platforms** SRX Series

This example shows how to configure VLANs, a Layer 2 interface, and an IRB interface that supports both IPv4 and IPv6 addresses. This example also shows how to configure

the device to use only ARP requests to learn the outgoing interfaces for unknown destination MAC addresses.



## Requirements

The device must be enabled for IPv6 flow processing. See "Flow-Based Processing for IPv6 Traffic on Security Devices" on page 68.

## Overview

This example creates the configuration described in Table 11 on page 71.

**Table 11: IPv6 Transparent Mode Configuration for IPv6 Flows**

| Feature | Name | Configuration Parameters |
|---|---|---|
| VLANs | vlan-a | VLAN 2 |
| | vlan-b | VLAN 10 |
| Logical interface | ge-0/0/0.0 | Trunk port for packets tagged with VLAN IDs 1 through 10 |
| Physical interface | ge-0/0/0 | VLAN ID 30 assigned to untagged packets |
| IRB interface | irb.0 | Addresses:<br>• IPv4 address 10.1.1.1/24<br>• IPv6 address 2001:0db8:2::1/64<br>Referenced in vlan-b VLAN |
| Learn the outgoing interfaces for unknown destination MAC addresses | | Use only ARP queries without traceroute requests |

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans vlan-a vlan-id 2
set vlans vlan-b vlan members 1-10
set interfaces ge-0/0/0 vlan-tagging native-vlan-id 30
set interfaces ge-0/0/0 unit 0 family ethernet-switching interface-mode trunk vlan
    members 1-10
set interfaces irb unit 0 family inet address 10.1.1.1/24
set interfaces irb unit 0 family inet6 address 2001:0db8::1/64
```

```
set vlans vlan-b l3-interface irb.0
set security flow ethernet-switching no-packet-flooding no-trace-route
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure transparent mode for IPv6 flows:

1. Configure VLANs.

```
[edit vlans]
user@host# set vlan-a vlan-id 2
user@host# set vlan-b vlan members 1-10
```

2. Configure the Layer 2 interface.

```
[edit interfaces ge-0/0/0]
user@host# set vlan-tagging native-vlan-id 30
user@host# set unit 0 family ethernet-switching interface-mode trunk vlan members
  1-10
```

3. Configure the IRB interface.

```
[edit interfaces irb unit 0]
user@host# set family inet address 10.1.1.1/24
user@host# set family inet6 address 2001:0db8::1/64
```

4. Configure the IRB interface for the VLAN.

```
[edit vlans]
user@host# set vlan-b l3-interface irb.0
```

5. Configure learning for unknown destination MAC addresses.

```
[edit security flow ethernet-switching]
user@host# set no-packet-flooding no-trace-route
```

## Results

From configuration mode, confirm your configuration by entering the **show vlans**, **show interfaces**, and **show security flow ethernet-switching** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show vlans
vlan-a {
  vlan-id 2;
}
  vlan-b {
    vlan members 1-10;
    l3-interface irb.0;
```

```
    }
user@host# show interfaces
ge-0/0/0 {
    vlan-tagging;
    native-vlan-id 30;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan members 1-10;
        }
    }
}
user@host# show security flow ethernet-switching
no-packet-flooding {
    no-trace-route;
}
```

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

## Verification

Confirm that the configuration is working properly.

- Verifying IPv6 Sessions on page 73
- Verifying IPv6 Gates on page 73
- Verifying IPv6 IP-action Settings on page 73

### Verifying IPv6 Sessions

**Purpose** Verify IPv6 sessions on the device.

**Action** From operational mode, enter the **show security flow session family inet6** command.

### Verifying IPv6 Gates

**Purpose** Verify IPv6 gates on the device.

**Action** From operational mode, enter the **show security flow gate family inet6** command.

### Verifying IPv6 IP-action Settings

**Purpose** Verify IPv6 IP-action settings on the device.

**Action** From operational mode, enter the **show security flow ip-action family inet6** command.

**Related Documentation**

- *Understanding IPv6 Address Space, Addressing, Address Format, and Address Types*
- Understanding IPv6 Flows in Transparent Mode on Security Devices on page 67

CHAPTER 11

# Configuring Secure Wire

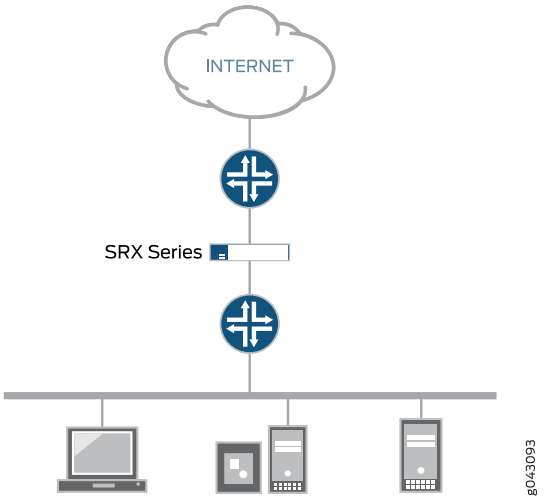


## Understanding Secure Wire on Security Devices

**Supported Platforms** SRX Series

Traffic that arrives on a specific interface can be forwarded unchanged through another interface. This mapping of interfaces, called secure wire, allows an SRX Series to be deployed in the path of network traffic without requiring a change to routing tables or a reconfiguration of neighboring devices. Figure 4 on page 76 shows a typical in-path deployment of an SRX Series with secure wire.

Figure 4: SRX Series In-Path Deployment with Secure Wire




Secure wire maps two peer interfaces. It differs from transparent and route modes in that there is no switching or routing lookup to forward traffic. As long as the traffic is permitted by a security policy, a packet arriving on one peer interface is immediately forwarded unchanged out of the other peer interface. There is no routing or switching decision made on the packet. Return traffic is also forwarded unchanged.

Secure wire mapping is configured with the **secure-wire** statement at the [**edit security forwarding-options**] hierarchy level; two Ethernet logical interfaces must be specified. The Ethernet logical interfaces must be configured with **family ethernet-switching** and each pair of interfaces must belong to the VLAN(s). The interfaces must be bound to security zones and a security policy configured to permit traffic between the zones.

This feature is available on Ethernet logical interfaces only; both IPv4 and IPv6 traffic are supported. You can configure interfaces for access or trunk mode. Secure wire supports chassis cluster redundant Ethernet interfaces. This feature does not support security features not supported in transparent mode, including NAT and IPsec VPN. Layer 7 features, including AppSecure, and IPS/IDP are supported.

NOTE: **Layer 7 feature, UTM is not supported in secure wire.**

Secure wire is a special case of Layer 2 transparent mode on SRX Series devices that provide point-to-point connections. This means that the two interfaces of a secure wire must ideally be directly connected to Layer 3 entities, such as routers or hosts. Secure wire interfaces can be connected to switches. However, note that a secure wire interface forwards all arriving traffic to the peer interface only if the traffic is permitted by a security policy.

Secure wire can coexist with Layer 3 mode. While you can configure Layer 2 and Layer 3 interfaces at the same time, traffic forwarding occurs independently on Layer 2 and Layer 3 interfaces.

Secure wire can coexist with Layer 2 transparent mode. If both features exist on the same SRX Series device, you need to configure them in different VLANs.

NOTE: Integrated routing and bridging (IRB) interfaces are not supported with secure wire.

**Related Documentation**

- Example: Simplifying SRX Series Device Deployment with Secure Wire over Access Mode Interfaces on page 77
- Example: Simplifying SRX Series Device Deployment with Secure Wire over Trunk Mode Interfaces on page 81
- Example: Simplifying SRX Series Device Deployment with Secure Wire over Aggregated Interface Member Links on page 84
- Example: Simplifying Chassis Cluster Deployment with Secure Wire over Redundant Ethernet Interfaces on page 89
- Example: Simplifying Chassis Cluster Deployment with Secure Wire over Aggregated Redundant Ethernet Interfaces on page 94
- Understanding Mixed Mode (Transparent and Route Mode) on Security Devices on page 11

## Example: Simplifying SRX Series Device Deployment with Secure Wire over Access Mode Interfaces

**Supported Platforms** SRX Series

If you are connecting an SRX Series device to other network devices, you can use secure wire to simplify the device deployment in the network. No changes to routing or forwarding tables on the SRX Series device and no reconfiguration of neighboring devices is needed. Secure wire allows traffic to be forwarded unchanged between specified access mode interfaces on an SRX Series device as long as it is permitted by security policies or other security features. Follow this example if you are connecting an SRX Series device to other network devices through access mode interfaces.

This example shows how to configure a secure wire mapping for two access mode interfaces. This configuration applies to scenarios where user traffic is not VLAN tagged.



### Requirements

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

## Overview

This example configures the secure wire access-sw that maps interface ge-0/0/0.0 to interface ge-0/0/1.0. The two peer interfaces are configured for access mode. The VLAN ID 10 is configured for the vlan-10 and the access mode interfaces.

NOTE: A specific VLAN ID must be configured for a VLAN.

### Topology

Figure 5 on page 78 shows the access mode interfaces that are mapped in secure wire access-sw.

**Figure 5: Secure Wire Access Mode Interfaces**


## Configuration

**CLI Quick Configuration**

NOTE: Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, some Layer 2 CLI configuration statements are enhanced, and some commands are changed. For detailed information about the modified hierarchies, see "Enhanced Layer 2 CLI Configuration Statement and Command Changes for Security Devices" on page 34.

The configuration statements shown below are for Junos OS Release 15.1X49-D10 or higher and Junos OS Release 17.3R1.

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans vlan-10 vlan-id 10
set interfaces ge-0/0/0 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/0 unit 0 family ethernet-switching vlan members 10
set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members 10
set security forwarding-options secure-wire access-sw interface [ge-0/0/0.0 ge-0/0/1.0]
set security zones security-zone trust interfaces ge-0/0/0.0
set security zones security-zone untrust interfaces ge-0/0/1.0
set security policies default-policy permit-all
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a secure wire mapping for access mode interfaces:

1. Configure the VLAN.

```
[edit vlans vlan-10]
user@host# set vlan-id 10
```

2. Configure the access mode interfaces.

```
[edit interfaces ]
user@host# set ge-0/0/0 unit 0 family ethernet-switching interface-mode access
user@host# set ge-0/0/1 unit 0 family ethernet-switching interface-mode access
user@host# set ge-0/0/0 unit 0 family ethernet-switching vlan members 10
user@host# set ge-0/0/1 unit 0 family ethernet-switching vlan members 10
```

3. Configure the secure wire mapping.

```
[edit security forwarding-options]
user@host# set secure-wire access-sw interface [ge-0/0/0.0 ge-0/0/1.0]
```

4. Configure security zones.

```
[edit security zones]
user@host# set security-zone trust interfaces ge-0/0/0.0
user@host# set security-zone untrust interfaces ge-0/0/1.0
```

5. Configure a security policy to permit traffic.

```
[edit security policies]
user@host# set default-policy permit-all
```

**Results**

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

```
user@host# show vlans
  vlan-10 {
    vlan-id 10;
    interface ge-0/0/0.0;
    interface ge-0/0/1.0;
  }
  user@host# show interfaces
  ge-0/0/0 {
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan {
          members 10;
```

```
        }
      }
    }
  }
  ge-0/0/1 {
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan {
          members 10;
        }
      }
    }
  }
  user@host# show security forwarding-options
  secure-wire {
    access-sw {
      interface [ ge-0/0/0.0 ge-0/0/1.0 ];
    }
  }
user@host# show security zones
  security-zone trust {
    interfaces {
      ge-0/0/0.0;
    }
  }
  security-zone untrust {
    interfaces {
      ge-0/0/1.0;
    }
  }
```

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

## Verification

Confirm that the configuration is working properly.

- Verifying Secure Wire Mapping on page 80
- Verifying the VLAN on page 81

### Verifying Secure Wire Mapping

**Purpose** Verify the secure wire mapping.

**Action** From operational mode, enter the **show security forwarding-options secure-wire** command.

```
user@host> show security forward-options secure-wire
Secure wire                    Interface       Link   Interface       Link

access-sw                      ge-0/0/0.0      up     ge-0/0/1.0      up
Total secure wires: 1
```

### Verifying the VLAN

**Purpose** Verify the VLAN.

**Action** From operational mode, enter the **show vlans vlan-10** command.

```
user@host> show vlans vlan-10
Routing instance        VLAN name             Tag     Interfaces
default-switch          vlan-10                10         ge-0/0/0.0
                                                          ge-0/0/1.0
```

**Related Documentation**
- Understanding Secure Wire on Security Devices on page 75

## Example: Simplifying SRX Series Device Deployment with Secure Wire over Trunk Mode Interfaces

**Supported Platforms** SRX Series

If you are connecting an SRX Series device to other network devices, you can use secure wire to simplify the device deployment in the network. No changes to routing or forwarding tables on the SRX Series device and no reconfiguration of neighboring devices is needed. Secure wire allows traffic to be forwarded unchanged between specified trunk mode interfaces on an SRX Series device as long as it is permitted by security policies or other security features. Follow this example if you are connecting an SRX Series device to other network devices through trunk mode interfaces.



### Requirements

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

### Overview

This example configures the secure wire trunk-sw that maps interface ge-0/1/0.0 to interface ge-0/1/1.0. The two peer interfaces are configured for trunk mode and carry user traffic tagged with VLAN IDs from 100 to 102. The VLAN ID list 100-102 is configured for the VLAN vlan-100 and the trunk mode interfaces.

NOTE: A specific VLAN ID must be configured for a VLAN.

### Topology

Figure 6 on page 82 shows the trunk mode interfaces that are mapped in secure wire trunk-sw.

Figure 6: Secure Wire Trunk Mode Interfaces


## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans vlan-100 vlan members 100-102
set interfaces ge-0/1/0 unit 0 family ethernet-switching interface-mode trunk vlan
  members 100-102
set interfaces ge-0/1/1 unit 0 family ethernet-switching interface-mode trunk vlan
  members 100-102
set security forwarding-options secure-wire trunk-sw interface [ge-0/1/0.0 ge-0/1/1.0]
set security zones security-zone trust interfaces ge-0/1/0.0
set security zones security-zone untrust interfaces ge-0/1/1.0
set security policies default-policy permit-all
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a secure wire mapping for trunk mode interfaces:

1. Configure the VLAN.

   ```
   [edit vlans vlan-100]
   user@host# set vlan members 100-102
   ```

2. Configure the trunk mode interfaces.

   ```
   [edit interfaces]
   user@host# set ge-0/1/0 unit 0 family ethernet-switching interface-mode trunk
     vlan members 100-102
   user@host# set ge-0/1/1 unit 0 family ethernet-switching interface-mode trunk
     vlan members 100-102
   ```

3. Configure the secure wire mapping.

   ```
   [edit security forwarding-options]
   user@host# set secure-wire trunk-sw interface [ge-0/1/0.0 ge-0/1/1.0]
   ```

4. Configure security zones.

```
[edit security zones]
user@host# set security-zone trust interfaces ge-0/1/0.0
user@host# set security-zone untrust interfaces ge-0/1/1.0
```

5. Configure a security policy to permit traffic.

```
[edit security policies]
user@host# set default-policy permit-all
```

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

```
user@host# show vlans
  vlan-100 {
    vlan members 100-102;
  }
  user@host# show interfaces
  ge-0/1/0 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan members 100-102;
      }
    }
  }
  ge-0/1/1 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan members 100-102;
      }
    }
  }
  user@host# show security forwarding-options
  secure-wire trunk-sw {
    interfaces [ge-0/1/0.0 ge-0/1/1.0];
  }
user@host# show security zones
  security-zone trust {
    interfaces {
      ge-0/1/0.0;
    }
  }
  security-zone untrust {
    interfaces {
      ge-0/1/1.0;
    }
  }
```

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

## Verification

Confirm that the configuration is working properly.


- Verifying Secure Wire Mapping on page 84
- Verifying the VLAN on page 84


### Verifying Secure Wire Mapping

**Purpose** Verify the secure wire mapping.

**Action** From operational mode, enter the **show security forwarding-options secure-wire** command.

```
user@host> show security forward-options secure-wire
Secure wire                    Interface       Link   Interface       Link

trunk-sw                       ge-0/1/0.0      up     ge-0/1/1.0      up
Total secure wires: 1
```

### Verifying the VLAN

**Purpose** Verify the VLAN.

**Action** From operational mode, enter the **show vlans** command.

```
user@host> show vlans
Routing instance        VLAN name             VLAN ID     Interfaces
default-switch          vlan-100-vlan-0100        100         ge-0/1/0.0
                                                            ge-0/1/1.0
default-switch          vlan-100-vlan-0101        101         ge-0/1/0.0
                                                            ge-0/1/1.0
default-switch          vlan-100-vlan-0102        102         ge-0/1/0.0
                                                            ge-0/1/1.0
```

NOTE: VLANs are automatically expanded, with one VLAN for each VLAN ID in the VLAN ID list.

**Related Documentation**


- Understanding Secure Wire on Security Devices on page 75


## Example: Simplifying SRX Series Device Deployment with Secure Wire over Aggregated Interface Member Links

**Supported Platforms** SRX Series

If you are connecting an SRX Series device to other network devices, you can use secure wire to simplify the device deployment in the network. No changes to routing or forwarding tables on the SRX Series device and no reconfiguration of neighboring devices is needed. Secure wire allows traffic to be forwarded unchanged between specified aggregated interface member links on an SRX Series device as long as it is permitted by security policies or other security features. Follow this example if you are connecting an SRX Series device to other network devices through aggregated interface member links.

NOTE: LACP is not supported. Secure wire mappings can be configured for member links of link bundles instead of directly mapping aggregated Ethernet interfaces.

NOTE: On SRX210, SRX220, SRX240, SRX300, SRX320, SRX340, SRX345, SRX550, and SRX650 devices, when you create an aggregated interface with two or more ports and set the family to Ethernet switching, and if a link in the bundle goes down, the traffic forwarded through the same link will be rerouted two seconds later. This causes an outage for the traffic being sent to the link until reroute is complete.

## Requirements

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

## Overview

This example configures secure wires for two aggregated Ethernet interface link bundles with two links each. Two separate secure wires ae-link1 and ae-link2 are configured using one link from each aggregated Ethernet link bundle. This static mapping requires that the two link bundles have the same number of links.

For link bundles, all logical interfaces of the secure wire mappings must belong to the same VLAN. VLAN ID 10 is configured for the VLAN vlan-10 and the logical interfaces. All logical interfaces of a link bundle must belong to the same security zone.

NOTE: A specific VLAN ID or VLAN ID list must be configured for a VLAN.

### Topology

Figure 7 on page 86 shows the aggregated interfaces that are mapped in secure wire configurations.

Figure 7: Secure Wire Aggregated Interfaces


## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans vlan-10 vlan-id 10
set interfaces ge-0/0/0 unit 0 family ethernet-switching interface-mode access vlan-id
  10
set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode access vlan-id
  10
set interfaces ge-0/1/0 unit 0 family ethernet-switching interface-mode access vlan-id
  10
set interfaces ge-0/1/1 unit 0 family ethernet-switching interface-mode access vlan-id
  10
set security forwarding-options secure-wire ae-link1-sw interface [ge-0/1/0.0 ge-0/1/1.0]
set security forwarding-options secure-wire ae-link2-sw interface [ge-0/0/0.0 ge-0/0/1.0]
set security zones security-zone trust interfaces ge-0/0/0.0
set security zones security-zone trust interfaces ge-0/1/0.0
set security zones security-zone untrust interfaces ge-0/0/1.0
set security zones security-zone untrust interfaces ge-0/1/1.0
set security policies default-policy permit-all
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a secure wire mapping for aggregated interface member links:

1. Configure the VLAN.

```
[edit vlans vlan-10]
user@host# set vlan-id10
```

2. Configure the interfaces.

```
[edit interfaces ]
user@host# set ge-0/0/0 unit 0 family ethernet-switching interface-mode access
  vlan-id 10
user@host# set ge-0/0/1 unit 0 family ethernet-switching interface-mode access
  vlan-id 10
user@host# set ge-0/1/0 unit 0 family ethernet-switching interface-mode access
  vlan-id 10
user@host# set ge-0/1/1 unit 0 family ethernet-switching interface-mode access
  vlan-id 10
```

3. Configure the secure wire mappings.

```
[edit security forwarding-options]
user@host# set secure-wire ae-link1-sw interface [ ge-0/1/0.0 ge-0/1/1.0 ]
user@host# set secure-wire ae-link2-sw interface [ ge-0/0/0.0 ge-0/0/1.0 ]
```

4. Configure security zones.

```
[edit security zones]
user@host# set security-zone trust interfaces ge-0/0/0.0
user@host# set security-zone trust interfaces ge-0/1/0.0
user@host# set security-zone untrust interfaces ge-0/0/1.0
user@host# set security-zone untrust interfaces ge-0/1/1.0
```

5. Configure a security policy to permit traffic.

```
[edit security policies]
user@host# set default-policy permit-all
```

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

```
user@host# show vlans
  vlan-10 {
    vlan-id 10;
  }
  user@host# show interfaces
  ge-0/0/0 {
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  ge-0/0/1 {
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  ge-0/1/0 {
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  ge-0/1/1{
```

```
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  user@host# show security forwarding-options
  secure-wire ae-link1-sw {
    interfaces [ge-0/1/0.0 ge-0/1/1.0];
  }
  secure-wire ae-link2-sw {
    interfaces [ge-0/0/0.0 ge-0/0/1.0];
  }
user@host# show security zones
  security-zone trust {
    interfaces {
      ge-0/0/0.0;
      ge-0/1/0.0;
    }
  }
  security-zone untrust {
    interfaces {
      ge-0/0/1.0;
      ge-0/1/1.0;
    }
  }
```

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

## Verification

Confirm that the configuration is working properly.

- Verifying Secure Wire Mapping on page 88
- Verifying the VLAN on page 89

### Verifying Secure Wire Mapping

**Purpose** Verify the secure wire mapping.

**Action** From operational mode, enter the **show security forwarding-options secure-wire** command.

```
user@host> show security forward-options secure-wire
Secure wire                    Interface       Link   Interface       Link

ae-link1-sw                    ge-0/1/0.0      up     ge-0/1/1.0      up
ae-link2-sw                    ge-0/0/0.0      up     ge-0/0/1.0      up
Total secure wires: 2
```

### Verifying the VLAN

**Purpose** Verify the VLAN.

**Action** From operational mode, enter the **show vlans vlan-10** command.

```
user@host> show vlans vlan-10
Routing instance        VLAN name            VLAN ID     Interfaces
default-switch          vlan-10                   10          ge-0/0/0.0
                                                              ge-0/0/1.0
                                                              ge-0/1/0.0
                                                              ge-0/1/1.0
```

**Related Documentation**
- Understanding Secure Wire on Security Devices on page 75

## Example: Simplifying Chassis Cluster Deployment with Secure Wire over Redundant Ethernet Interfaces

**Supported Platforms** SRX Series

If you are connecting an SRX Series chassis cluster to other network devices, you can use secure wire to simplify the cluster deployment in the network. No changes to routing or forwarding tables on the cluster and no reconfiguration of neighboring devices is needed. Secure wire allows traffic to be forwarded unchanged between specified redundant Ethernet interfaces on the SRX Series chassis cluster as long as it is permitted by security policies or other security features. Follow this example if you are connecting an SRX Series chassis cluster to other network devices through redundant Ethernet interfaces.



### Requirements

Before you begin:

- Connect a pair of the same SRX Series devices in a chassis cluster.
- Configure the chassis cluster node ID and cluster ID.
- Set the number of redundant Ethernet interfaces in the chassis cluster.
- Configure the chassis cluster fabric.
- Configure chassis cluster redundancy group (in this example redundancy group 1 is used).

For more information, see the *Chassis Cluster Feature Guide for SRX Series Devices*.

## Overview

Secure wire is supported over redundant Ethernet interfaces in a chassis cluster. The two redundant Ethernet interfaces must be configured in the same redundancy group. If failover occurs, both redundant Ethernet interfaces must fail over together.

NOTE: Secure wire mapping of redundant Ethernet link aggregation groups (LAGs) are not supported. LACP is not supported.

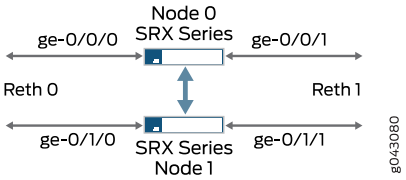
This example configures the secure wire reth-sw that maps ingress interface reth0.0 to egress interface reth1.0. Each redundant Ethernet interface consists of two child interfaces, one on each node of the chassis cluster. The two redundant Ethernet interfaces are configured for access mode. VLAN ID 10 is configured for the VLAN vlan-10 and the redundant Ethernet interfaces.

NOTE: A specific VLAN ID or VLAN ID list must be configured for a VLAN.

### Topology

Figure 8 on page 90 shows the redundant Ethernet interfaces that are mapped in secure wire reth-sw.

Figure 8: Secure Wire Redundant Ethernet Interfaces




## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans vlan-10 vlan-id 10
set interfaces ge-0/0/0 gigether-options redundant-parent reth0
set interfaces ge-0/0/1 gigether-options redundant-parent reth1
set interfaces ge-0/1/0 gigether-options redundant-parent reth0
set interfaces ge-0/1/1 gigether-options redundant-parent reth1
set interfaces reth0 unit 0 family ethernet-switching interface-mode access vlan-id 10
set interfaces reth1 unit 0 family ethernet-switching interface-mode access vlan-id 10
set interfaces reth0 redundant-ether-options redundancy-group 1
```

```
set interfaces reth1 redundant-ether-options redundancy-group 1
set security forwarding-options secure-wire reth-sw interface [reth0.0 reth1.0]
set security zones security-zone trust interfaces reth0.0
set security zones security-zone untrust interfaces reth1.0
set security policies default-policy permit-all
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a secure wire mapping for chassis cluster redundant Ethernet interfaces:

1. Configure the VLAN.

```
[edit vlans vlan-10]
user@host# set vlan-id 10
```

2. Configure the redundant Ethernet interfaces.

```
[edit interfaces ]
user@host# set ge-0/0/0 gigether-options redundant-parent reth0
user@host# set ge-0/0/1 gigether-options redundant-parent reth1
user@host# set ge-0/1/0 gigether-options redundant-parent reth0
user@host# set ge-0/1/1 gigether-options redundant-parent reth1

user@host#set reth0 unit 0 family ethernet-switching interface-mode access vlan-id
  10
user@host#set reth1 unit 0 family ethernet-switching interface-mode access vlan-id
  10

user@host# set reth0 redundant-ether-options redundancy-group 1
user@host# set reth1 redundant-ether-options redundancy-group 1
```

3. Configure the secure wire mapping.

```
[edit security forwarding-options]
user@host# set secure-wire reth-sw interface [reth0.0 reth1.0]
```

4. Configure security zones.

```
[edit security zones]
user@host# set security-zone trust interfaces reth0.0
user@host# set security-zone untrust interfaces reth1.0
```

5. Configure a security policy to permit traffic.

```
[edit security policies]
user@host# set default-policy permit-all
```

**Results**

From configuration mode, confirm your configuration by entering the **show vlans**, **show interfaces**, **show security forwarding-options**, and **show security zones** commands. If the

output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show vlans
  vlan-10 {
    vlan-id 10;
  }
  user@host# show interfaces
  ge-0/0/0 {
    gigether-options {
      redundant-parent reth0;
    }
  }
  ge-0/0/1 {
    gigether-options {
      redundant-parent reth1;
    }
  }
  ge-0/1/0 {
    gigether-options {
      redundant-parent reth0;
    }
  }
  ge-0/1/1 {
    gigether-options {
      redundant-parent reth1;
    }
  }
  reth0 {
    redundant-ether-options {
      redundancy-group 1;
    }
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  reth1 {
    redundant-ether-options {
      redundancy-group 1;
    }
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  user@host# show security forwarding-options
  secure-wire reth-sw {
    interfaces [reth0.0 reth1.0];
  }
user@host# show security zones
  security-zone trust {
```

```
    interfaces {
      reth0.0;
    }
  }
  security-zone untrust {
    interfaces {
      reth1.0;
    }
  }
```

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

## Verification

Confirm that the configuration is working properly.

- Verifying Secure Wire Mapping on page 93
- Verifying the VLAN on page 93

### Verifying Secure Wire Mapping

**Purpose** Verify the secure wire mapping.

**Action** From operational mode, enter the **show security forwarding-options secure-wire** command.

```
user@host> show security forward-options secure-wire
node0:
--------------------------------------------------------------------------
Secure wire                       Interface       Link   Interface       Link

reth-sw                           reth0.0         up     reth1.0         up

Total secure wires: 1

node1:
--------------------------------------------------------------------------
Secure wire                       Interface       Link   Interface       Link

reth-sw                           reth0.0         up     reth1.0         up

Total secure wires: 1
```

### Verifying the VLAN

**Purpose** Verify the VLAN.

**Action** From operational mode, enter the **show vlan vlan-10** command.

```
user@host> show vlan vlan-10
Routing instance        VLAN Name             VLAN ID     Interfaces
default-switch          vlan-10                    10         reth0.0
                                                              reth1.0
```

**Related Documentation**

- Understanding Secure Wire on Security Devices on page 75
- Example: Simplifying Chassis Cluster Deployment with Secure Wire over Aggregated Redundant Ethernet Interfaces on page 94

## Example: Simplifying Chassis Cluster Deployment with Secure Wire over Aggregated Redundant Ethernet Interfaces

**Supported Platforms** SRX Series

If you are connecting an SRX Series chassis cluster to other network devices, you can use secure wire to simplify the cluster deployment in the network. No changes to routing or forwarding tables on the cluster and no reconfiguration of neighboring devices is needed. Secure wire allows traffic to be forwarded unchanged between specified redundant Ethernet interfaces on the SRX Series chassis cluster as long as it is permitted by security policies or other security features. Follow this example if you are connecting an SRX Series chassis cluster to other network devices through aggregated redundant Ethernet interfaces.

NOTE: Secure wires cannot be configured for redundant Ethernet interface link aggregation groups (LAGs). For the secure wire mapping shown in this example, there is no LAG configuration on the SRX Series chassis cluster. Each redundant Ethernet interface consists of two child interfaces, one on each node of the chassis cluster. Users on upstream or downstream devices connected to the SRX Series cluster can configure the redundant Ethernet interface child links in LAGs.



### Requirements

Before you begin:

- Connect a pair of the same SRX Series devices in a chassis cluster.
- Configure the chassis cluster node ID and cluster ID.
- Set the number of redundant Ethernet interfaces in the chassis cluster.
- Configure the chassis cluster fabric.
- Configure the chassis cluster redundancy group (in this example, redundancy group 1 is used).

For more information, see the *Chassis Cluster Feature Guide for SRX Series Devices*.

## Overview

This example configures secure wires for four redundant Ethernet interfaces: reth0, reth1, reth2, and reth3. Each redundant Ethernet interface consists of two child interfaces, one on each node of the chassis cluster. All four redundant Ethernet interfaces must be in the same VLAN—in this example, the VLAN is vlan-0. Two of the redundant Ethernet interfaces, reth0.0 and reth2.0, are assigned to the trust zone, while the other two interfaces, reth1.0 and reth3.0, are assigned to the untrust zone.

This example configures the following secure wires:

- reth-sw1 maps interface reth0.0 to interface reth1.0
- reth-sw2 maps interface reth2.0 to reth3.0

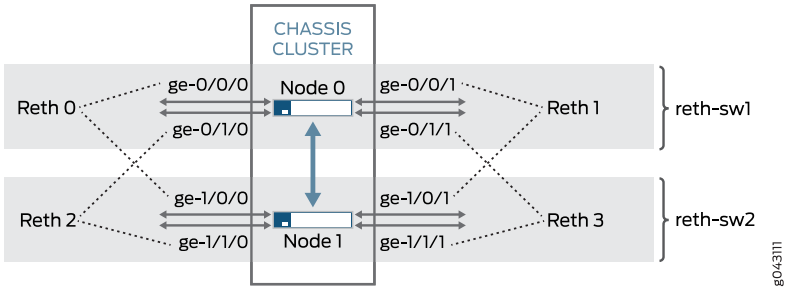
All redundant Ethernet interfaces are configured for access mode. VLAN ID 10 is configured for the VLAN vlan-0 and the redundant Ethernet interfaces.

NOTE: A specific VLAN ID or VLAN ID list must be configured for a VLAN.

### Topology

Figure 9 on page 95 shows the redundant Ethernet interface child links that are mapped in secure wire configurations reth-sw1 and reth-sw2. Each redundant Ethernet interface consists of two child interfaces, one on each node of the chassis cluster.

**Figure 9: Secure Wire Redundant Ethernet Interface Child Links**




Users on upstream or downstream devices connected to the SRX Series cluster can configure redundant Ethernet interface child links in a LAG as long as the LAG does not span chassis cluster nodes. For example, ge-0/0/0 and ge-0/1/0 and ge-0/0/1 and ge-0/1/1 on node 0 can be configured as LAGs on connected devices. In the same way, ge-1/0/0 and ge-1/1/0 and ge-1/0/1 and ge-1/1/1 on node 1 can be configured as LAGs on connected devices.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans vlan-0 vlan-id 10
set interfaces ge-0/0/0 gigether-options redundant-parent reth0
set interfaces ge-0/0/1 gigether-options redundant-parent reth1
set interfaces ge-0/1/0 gigether-options redundant-parent reth2
set interfaces ge-0/1/1 gigether-options redundant-parent reth3
set interfaces ge-1/0/0 gigether-options redundant-parent reth0
set interfaces ge-1/0/1 gigether-options redundant-parent reth1
set interfaces ge-1/1/0 gigether-options redundant-parent reth2
set interfaces ge-1/1/1 gigether-options redundant-parent reth3
set interfaces reth0 unit 0 family ethernet-switching interface-mode access vlan-id 10
set interfaces reth1 unit 0 family ethernet-switching interface-mode access vlan-id 10
set interfaces reth2 unit 0 family ethernet-switching interface-mode access vlan-id 10
set interfaces reth3 unit 0 family ethernet-switching interface-mode access vlan-id 10
set interfaces reth0 redundant-ether-options redundancy-group 1
set interfaces reth1 redundant-ether-options redundancy-group 1
set interfaces reth2 redundant-ether-options redundancy-group 1
set interfaces reth3 redundant-ether-options redundancy-group 1
set security forwarding-options secure-wire reth-sw1 interface [reth0.0 reth1.0]
set security forwarding-options secure-wire reth-sw2 interface [reth2.0 reth3.0]
set security zones security-zone trust interfaces reth0.0
set security zones security-zone trust interfaces reth2.0
set security zones security-zone untrust interfaces reth1.0
set security zones security-zone untrust interfaces reth3.0
set security policies default-policy permit-all
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a secure wire mapping for aggregated interface member links:

1. Configure the VLAN.

```
[edit vlans vlan-0]
user@host# set vlan-id 10
```

2. Configure the redundant Ethernet interfaces.

```
[edit interfaces ]
user@host# set ge-0/0/0 gigether-options redundant-parent reth0
user@host# set ge-0/0/1 gigether-options redundant-parent reth1
user@host# set ge-0/1/0 gigether-options redundant-parent reth2
user@host# set ge-0/1/1 gigether-options redundant-parent reth3
user@host# set ge-1/0/0 gigether-options redundant-parent reth0
user@host# set ge-1/0/1 gigether-options redundant-parent reth1
user@host# set ge-1/1/0 gigether-options redundant-parent reth2
user@host# set ge-1/1/1 gigether-options redundant-parent reth3

user@host# set reth0 unit 0 family ethernet-switching interface-mode access
  vlan-id 10
```

```
user@host# set reth1 unit 0 family ethernet-switching interface-mode access vlan-id
  10
user@host# set reth2 unit 0 family ethernet-switching interface-mode access vlan-id
  10
user@host# set reth3 unit 0 family ethernet-switching interface-mode access vlan-id
  10


user@host# set reth0 redundant-ether-options redundancy-group 1
user@host# set reth1 redundant-ether-options redundancy-group 1
user@host# set reth2 redundant-ether-options redundancy-group 1
user@host# set reth3 redundant-ether-options redundancy-group 1
```

3. Configure the secure wire mappings.

```
[edit security forwarding-options]
user@host# set secure-wire reth-sw1 interface [reth0.0 reth1.0]
user@host# set secure-wire reth-sw2 interface [reth2.0 reth3.0]
```

4. Configure security zones.

```
[edit security zones]
user@host# set security-zone trust interfaces reth0.0
user@host# set security-zone trust interfaces reth2.0


user@host# set security-zone untrust interfaces reth1.0
user@host# set security-zone untrust interfaces reth3.0
```

5. Configure a security policy to permit traffic.

```
[edit security policies]
user@host# set default-policy permit-all
```

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

```
user@host# show vlans
  vlan-0 {
    vlan-id 10;
  }
  user@host# show interfaces
  ge-0/0/0 {
    gigether-options {
      redundant-parent reth0;
    }
  }
  ge-0/0/1 {
    gigether-options {
      redundant-parent reth1;
    }
  }
```

```
ge-0/1/0 {
    gigether-options {
        redundant-parent reth2;
    }
}
ge-0/1/1 {
    gigether-options {
        redundant-parent reth3;
    }
}
ge-1/0/0 {
    gigether-options {
        redundant-parent reth0;
    }
}
ge-1/0/1 {
    gigether-options {
        redundant-parent reth1;
    }
}
ge-1/1/0 {
    gigether-options {
        redundant-parent reth2;
    }
}
ge-1/1/1 {
    gigether-options {
        redundant-parent reth3;
    }
}
reth0 {
    redundant-ether-options {
        redundancy-group 1;
    }
    unit 0 {
        family ethernet-switching {
            interface-mode access;
            vlan-id 10;
        }
    }
}
reth1 {
    redundant-ether-options {
        redundancy-group 1;
    }
    unit 0 {
        family ethernet-switching {
            interface-mode access;
            vlan-id 10;
        }
    }
}
reth2 {
    redundant-ether-options {
        redundancy-group 1;
    }
```

```
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  reth3 {
    redundant-ether-options {
      redundancy-group 1;
    }
    unit 0 {
      family ethernet-switching {
        interface-mode access;
        vlan-id 10;
      }
    }
  }
  user@host# show security forwarding-options
  secure-wire reth-sw1 {
    interfaces [reth0.0 reth1.0];
  }
  secure-wire reth-sw2 {
    interfaces [reth2.0 reth3.0];
  }
user@host# show security zones
  security-zone trust {
    interfaces {
      reth0.0;
      reth2.0;
    }
  }
  security-zone untrust {
    interfaces {
      reth1.0;
      reth3.0;
    }
  }
```

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

## Verification

Confirm that the configuration is working properly.

- Verifying Secure Wire Mapping on page 99
- Verifying VLAN on page 100

### Verifying Secure Wire Mapping

**Purpose** Verify the secure wire mapping.

**Action** From operational mode, enter the **show security forwarding-options secure-wire** command.

```
user@host> show security forward-options secure-wire
node0:
--------------------------------------------------------------------------
Secure wire                        Interface       Link   Interface       Link

reth-sw1                           reth0.0         up     reth1.0         up
reth-sw2                           reth2.0         up     reth3.0         up

Total secure wires: 2

node1:
--------------------------------------------------------------------------
Secure wire                        Interface       Link   Interface       Link

reth-sw1                           reth0.0         up     reth1.0         up
reth-sw2                           reth2.0         up     reth3.0         up

Total secure wires: 2
```

## Verifying VLAN

**Purpose** Verify the VLAN.

**Action** From operational mode, enter the **show vlans vlan-0** command.

```
user@host> show vlans vlan-0
Routing instance        VLAN name            VLAN ID     Interfaces
default-switch          vlan-0                    10
                                                          reth0.0
                                                          reth1.0
                                                          reth2.0
                                                          reth3.0
```

**Related Documentation**

- Understanding Secure Wire on Security Devices on page 75
- Example: Simplifying Chassis Cluster Deployment with Secure Wire over Redundant Ethernet Interfaces on page 89

PART 3

# Configuring Ethernet Ports for Switching

CHAPTER 12

# Configuring Switching Modes



## Understanding Switching Modes on Security Devices

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345

There are two types of switching modes:

- Switching Mode–The uPIM appears in the list of interfaces as a single interface, which is the first interface on the uPIM. For example, ge-2/0/0. You can optionally configure each uPIM port only for autonegotiation, speed, and duplex mode. A uPIM in switching mode can perform the following functions:
  - Layer 3 forwarding—Routes traffic destined for WAN interfaces and other PIMs present on the chassis.
  - Layer 2 forwarding—Switches intra-LAN traffic from one host on the LAN to another LAN host (one port of uPIM to another port of same uPIM).
- Enhanced Switching Mode–Each port can be configured for switching or routing mode. This usage differs from the routing and switching modes, in which all ports must be in either switching or routing mode. The uPIM in enhanced switching mode provides the following features:
  - Supports configuration of different types of VLANs and inter-VLAN routing.
  - Supports Layer 2 control plane protocol such as Link Aggregation Control Protocol (LACP).
  - Supports port-based Network Access Control (PNAC) by means of authentication servers.

NOTE: The SRX300 and SRX320 devices support enhanced switching mode only. When you set a multiport uPIM to enhanced switching mode, all the Layer 2 switching features are supported on the uPIM. (Platform support depends on the Junos OS release in your installation.)

You can set a multiport Gigabit Ethernet uPIM on a device to either switching or enhanced switching mode.

When you set a multiport uPIM to switching mode, the uPIM appears as a single entity for monitoring purposes. The only physical port settings that you can configure are autonegotiation, speed, and duplex mode on each uPIM port, and these settings are optional.

**Related Documentation**

- Example: Configuring Switching Modes on Security Devices on page 111
- Ethernet Ports Switching Overview for Security Devices on page 104

## Ethernet Ports Switching Overview for Security Devices

**Supported Platforms** SRX Series

Certain ports on Juniper Networks devices can function as Ethernet access switches that switch traffic at Layer 2 and route traffic at Layer 3.

You can deploy supported devices in branch offices as an access or desktop switch with integrated routing capability, thus eliminating intermediate access switch devices from your network topology. The Ethernet ports provide switching while the Routing Engine provides routing functionality, enabling you to use a single device to provide routing, access switching, and WAN interfaces.

This topic contains the following sections:



### Supported Devices and Ports

Juniper Networks supports switching features on a variety of Ethernet ports and devices (see Table 12 on page 104). Platform support depends on the Junos OS release in your installation. The following ports and devices are included:

- Onboard Ethernet ports (Gigabit and Fast Ethernet built-in ports) on the SRX300, SRX320, SRX320 PoE, SRX340, SRX345, SRX550M and SRX1500 devices.
- Multiport Gigabit Ethernet XPIM on the SRX650 device.

**Table 12: Supported Devices and Ports for Switching Features**

| Device | Ports |
|---|---|
| SRX100 devices | Onboard Fast Ethernet ports (fe-0/0/0 and fe-0/0/7) |

Table 12: Supported Devices and Ports for Switching Features *(continued)*

| Device | Ports |
|---|---|
| SRX210 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 and ge-0/0/1) and 1-Port Gigabit Ethernet SFP Mini-PIM port.<br><br>Onboard Fast Ethernet ports (fe-0/0/2 and fe-0/0/7) |
| SRX220 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 through ge-0/0/7) and 1-Port Gigabit Ethernet SFP Mini-PIM port. |
| SRX240 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 through ge-0/0/15) and 1-Port Gigabit Ethernet SFP Mini-PIM port. |
| SRX300 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 and ge-0/0/7) |
| SRX320 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 and ge-0/0/7) |
| SRX340 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 through ge-0/0/15) |
| SRX345 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 through ge-0/0/15) |
| SRX550 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 through ge-0/0/9, Multiport Gigabit Ethernet XPIM modules, and 1-Port Gigabit Ethernet SFP Mini-PIM port. |
| SRX550M devices | Onboard Gigabit Ethernet ports (ge-0/0/0 through ge-0/0/9 and Multiport Gigabit Ethernet XPIM modules. |
| SRX650 devices | Multiport Gigabit Ethernet XPIM modules<br><br>NOTE: On SRX650 devices, Ethernet switching is not supported on Gigabit Ethernet interfaces (ge-0/0/0 through ge-0/0/3 ports). |
| SRX1500 devices | Onboard Gigabit Ethernet ports (ge-0/0/0 through ge-0/0/19) |

On the SRX100. SRX220, SRX240, SRX300, SRX320, SRX340 and SRX345 devices, you can set the onboard Gigabit Ethernet ports to operate as either switched ports or routed ports. (Platform support depends on the Junos OS release in your installation.)

## Integrated Bridging and Routing

Integrated bridging and routing (IRB) provides support for simultaneous Layer 2 swiching and Layer 3 routing within the same VLAN. Packets arriving on an interface of the VLAN are switched or routed based on the destination MAC address of the packet. Packets with the router's MAC address as the destination are routed to other Layer 3 interfaces.

## Link Layer Discovery Protocol and LLDP-Media Endpoint Discovery

Devices use Link Layer Discovery Protocol (LLDP) and LLDP-Media Endpoint Discovery (MED) to learn and distribute device information about network links. The information

allows the device to quickly identify a variety of systems, resulting in a LAN that interoperates smoothly and efficiently.

LLDP-capable devices transmit information in Type Length Value (TLV) messages to neighbor devices. Device information can include specifics, such as chassis and port identification and system name and system capabilities. The TLVs leverage this information from parameters that have already been configured in the Junos OS.

LLDP-MED goes one step further, exchanging IP-telephony messages between the device and the IP telephone. These TLV messages provide detailed information about Power over Ethernet (PoE) policy. The PoE Management TLVs let the device ports advertise the power level and power priority needed. For example, the device can compare the power needed by an IP telephone running on a PoE interface with available resources. If the device cannot meet the resources required by the IP telephone, the device could negotiate with the telephone until a compromise on power is reached.

The following basic TLVs are supported:

- Chassis Identifier—The MAC address associated with the local system.
- Port identifier—The port identification for the specified port in the local system.
- Port Description—The user-configured port description. The port description can be a maximum of 256 characters.
- System Name—The user-configured name of the local system. The system name can be a maximum of 256 characters.
- Switching Features Overview—This information is not configurable, but taken from the software.
- System Capabilities—The primary function performed by the system. The capabilities that system supports; for example, Ethernet switching or router. This information is not configurable, but based on the model of the product.
- Management Address—The IP management address of the local system.

The following LLDP-MED TLVs are supported:

- LLDP-MED Capabilities—A TLV that advertises the primary function of the port. The values range from 0 through 15:
  - 0—Capabilities
  - 1—Network policy
  - 2—Location identification
  - 3—Extended power through medium-dependent interface power-sourcing equipment (MDI-PSE)
  - 4—Inventory
  - 5–15—Reserved
- LLDP-MED Device Class Values:

- 0—Class not defined
- 1—Class 1 device
- 2—Class 2 device
- 3—Class 3 device
- 4—Network connectivity device
- 5–255— Reserved

NOTE: Starting in Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, Link Layer Discovery Protocol (LLDP) and LLDP-Media Endpoint Discovery (MFD) are enabled on SRX300, SRX320, SRX340, SRX345, SRX550M and SRX1500 devices.

- Network Policy—A TLV that advertises the port VLAN configuration and associated Layer 2 and Layer 3 attributes. Attributes include the policy identifier, application types, such as voice or streaming video, 802.1Q VLAN tagging, and 802.1p priority bits and Diffserv code points.
- Endpoint Location—A TLV that advertises the physical location of the endpoint.
- Extended Power via MDI—A TLV that advertises the power type, power source, power priority, and power value of the port. It is the responsibility of the PSE device (network connectivity device) to advertise the power priority on a port.

LLDP and LLDP-MED must be explicitly configured on uPIMs (in enhanced switching mode) on base ports on SRX100, SRX210, SRX240, SRX300, SRX320, SRX340, and SRX345 devices, and Gigabit Backplane Physical Interface Modules (GPIMs) on SRX650 devices. (Platform support depends on the Junos OS release in your installation.) To configure LLDP on all interfaces or on a specific interface, use the **lldp** statement at the [**set protocols**] hierarchy level. To configure LLDP-MED on all interfaces or on a specific interface, use the **lldp-med** statement at the [**set protocols**] hierarchy level.

## Types of Switch Ports

The ports, or interfaces, on a switch operate in either access mode or trunk mode.

An interface in access mode connects to a network device, such as a desktop computer, an IP telephone, a printer, a file server, or a security camera. The interface itself belongs to a single VLAN. The frames transmitted over an access interface are normal Ethernet frames.

Trunk interfaces handle traffic for multiple VLANs, multiplexing the traffic for all those VLANs over the same physical connection. Trunk interfaces are generally used to interconnect switches to one another.

## uPIM in a Daisy Chain

You cannot combine multiple uPIMs to act as a single integrated switch. However, you can connect uPIMs on the same chassis externally by physically connecting a port on one uPIM to a port on another uPIM in a daisy-chain fashion.

Two or more uPIMs daisy-chained together create a single switch with a higher port count than either individual uPIM. One port on each uPIM is used solely for the connection. For example, if you daisy-chain a 6-port uPIM and an 8-port uPIM, the result operates as a 12-port uPIM. Any port of a uPIM can be used for daisy chaining.

Configure the IP address for only one of the daisy-chained uPIMs, making it the primary uPIM. The secondary uPIM routes traffic to the primary uPIM, which forwards it to the Routing Engine. This results in some increase in latency and packet drops due to oversubscription of the external link.

Only one link between the two uPIMs is supported. Connecting more than one link between uPIMs creates a loop topology, which is not supported.

## Q-in-Q VLAN Tagging

Q-in-Q tunneling, defined by the IEEE 802.1ad standard, allows service providers on Ethernet access networks to extend a Layer 2 Ethernet connection between two customer sites.

In Q-in-Q tunneling, as a packet travels from a customer VLAN (C-VLAN) to a service provider's VLAN, a service provider-specific 802.1Q tag is added to the packet. This additional tag is used to segregate traffic into service-provider-defined service VLANs (S-VLANs). The original customer 802.1Q tag of the packet remains and is transmitted transparently, passing through the service provider's network. As the packet leaves the S-VLAN in the downstream direction, the extra 802.1Q tag is removed.

NOTE: When Q-in-Q tunneling is configured for a service provider's VLAN, all Routing Engine packets, including packets from the routed VLAN interface, that are transmitted from the customer-facing access port of that VLAN will always be untagged.

There are three ways to map C-VLANs to an S-VLAN:

- All-in-one bundling—Use the **dot1q-tunneling** statement at the [**edit vlans**] hierarchy level to map without specifying customer VLANs. All packets from a specific access interface are mapped to the S-VLAN.
- Many-to-one bundling—Use the **customer-vlans** statement at the [**edit vlans**] hierarchy level to specify which C-VLANs are mapped to the S-VLAN.
- Mapping C-VLAN on a specific interface—Use the **mapping** statement at the [**edit vlans**] hierarchy level to map a specific C-VLAN on a specified access interface to the S-VLAN.

Table 13 on page 109 lists the C-VLAN to S-VLAN mapping supported on SRX Series devices. (Platform support depends on the Junos OS release in your installation.)

Table 13: Supported Mapping Methods

| Mapping | SRX210 | SRX240 | SRX300 | SRX320 | SRX340 | SRX345 | SRX550M | SRX650 |
|---|---|---|---|---|---|---|---|---|
| All-in-one bundling | Yes | Yes | No | No | Yes | Yes | Yes | Yes |
| Many-to-one bundling | No | No | No | No | Yes | Yes | Yes | Yes |
| Mapping C-VLAN on a specific interface | No | No | No | No | Yes | Yes | Yes | Yes |

NOTE: VLAN translation is supported on SRX300 and SRX320 devices and these devices do not support Q-in-Q tunneling.

NOTE: On SRX650 devices, in the dot1q-tunneling configuration options, customer VLANs range and VLAN push do not work together for the same S-VLAN, even when you commit the configuration. If both are configured, then VLAN push takes priority over customer VLANs range.

IRB interfaces are supported on Q-in-Q VLANs for SRX210, SRX240, SRX340, SRX345, and SRX650 devices. Packets arriving on an IRB interface on a Q-in-Q VLAN are routed regardless of whether the packet is single or double tagged. The outgoing routed packets contain an S-VLAN tag only when exiting a trunk interface; the packets exit the interface untagged when exiting an access interface. (Platform support depends on the Junos OS release in your installation.)

In a Q-in-Q deployment, customer packets from downstream interfaces are transported without any changes to source and destination MAC addresses. You can disable MAC address learning at both the interface level and the VLAN level. Disabling MAC address learning on an interface disables learning for all the VLANs of which that interface is a member. When you disable MAC address learning on a VLAN, MAC addresses that have already been learned are flushed.

On SRX100, SRX210, SRX240, SRX300, SRX320, SRX340, SRX345, and SRX650 devices (with platform support depending on the Junos OS release in your installation), on the Layer 3 aggregated Ethernet, the following features are not supported:

- Encapsulations (such as CCC, VLAN CCC, VPLS, and PPPoE)
- J-Web

- On all SRX Series devices, the Link Layer Discovery Protocol (LLDP) is not supported on redundant Ethernet (reth) interfaces.
- On SRX550M devices the aggregate Ethernet (ae) interface with XE member interface cannot be configured with the Ethernet switching family.
- On SRX300, SRX320, SRX340, SRX345, and SRX550M devices, the Q-in-Q support on a Layer 3 interface has the following limitations:
  - Double tagging is not supported on reth and ae interfaces.
  - Multitopology routing is not supported in flow mode and in chassis clusters.
  - Dual tagged frames are not supported on encapsulations (such as CCC, TCC, VPLS, and PPPoE)
  - On Layer 3 logical interfaces, **input-vlan-map**, **output-vlan-map**, **inner-range**, and **inner-list** are not applicable
  - Only TPIDs with 0x8100 are supported, and the maximum number of tags is 2.
  - Dual tagged frames are accepted only for logical interfaces with IPV4 and IPv6 families.
- On SRX100, SRX210, SRX240, SRX300, SRX320, SRX340, SRX345, and SRX650 devices (with platform support depending on the Junos OS release in your installation), on the routed VLAN interface (RVI), the following features are not supported:
  - IS-IS (family ISO)
  - Encapsulations (Ether CCC, VLAN CCC, VPLS, PPPoE, and so on) on VLAN interfaces
  - CLNS
  - DVMRP
  - VLAN interface MAC change
  - G-ARP
  - Change VLAN-Id for VLAN interface

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D60 | Starting in Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, Link Layer Discovery Protocol (LLDP) and LLDP-Media Endpoint Discovery (MFD) are enabled on SRX300, SRX320, SRX340, SRX345, SRX550M and SRX1500 devices. |

**Related Documentation**

- Understanding Switching Modes on Security Devices on page 103

## Example: Configuring Switching Modes on Security Devices

**Supported Platforms** SRX300, SRX320, SRX340, SRX345



### Requirements

Before you begin, see "Ethernet Ports Switching Overview for Security Devices" on page 104.

### Overview

In this example, you configure **chassis** and set the l2-learning protocol to global mode switching. You then set a physical port parameter on the l2-learning protocols.

### 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set protocols l2-learning global-mode switching
set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode access
```

**Step-by-Step Procedure**

To configure switching mode:

1. Set l2-learning protocol to global mode switching.

   ```
   [edit protocols l2-learning]
   user@host# set protocols l2-learning global-mode switching
   ```

2. Set a physical port parameter on the l2-learning protocols.

   ```
   [edit]
   user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode
     access
   ```

3. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

**Results**

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

```
[edit]
user@host# show protocols
l2-learning {
  global-mode switching;
}
```

```
[edit]
user@host# show interfaces
ge-0/0/1 {
  unit 0 {
    family ethernet-switching {
      interface-mode access;
    }
  }
}
```

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

## Verification

Confirm that the configuration is working properly.

- Verifying the Switching Mode on page 112
- Verifying the Ethernet switching on Interface ge-0/0/1 on page 113

### Verifying the Switching Mode

**Purpose** Make sure that the switching mode is configured as expected.

**Action** From operational mode, enter the **show ethernet-switching global-information** command.

```
user@host> show ethernet-switching global-information
```

```
Global Configuration:

MAC aging interval    : 300
MAC learning          : Enabled
MAC statistics        : Disabled
MAC limit Count       : 16383
MAC limit hit         : Disabled
MAC packet action drop: Disabled
MAC+IP aging interval : IPv4 - 1200 seconds
                        IPv6 - 1200 seconds
MAC+IP limit Count    : 393215
MAC+IP limit reached  : No
LE  aging time        : 1200
LE  VLAN aging time   : 1200
Global Mode           : Switching
RE state              : Master
```

**Meaning** The sample output shows that the global mode switching is configured as expected.

### Verifying the Ethernet switching on Interface ge-0/0/1

**Purpose** Make sure that the Ethernet switching is configured as expected on interface ge-0/0/1.

**Action** From operational mode, enter the **show interfaces ge-0/0/1 brief** command.

```
user@host> show interfaces ge-0/0/1 brief
```

```
Physical interface: ge-0/0/1, Enabled, Physical link is Down
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 1000mbps, Loopback:
 Disabled, Source filtering: Disabled, Flow control: Disabled, Auto-negotiation:
 Enabled, Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None

  Logical interface ge-0/0/1.0
    Flags: Device-Down SNMP-Traps 0x0 Encapsulation: Ethernet-Bridge
    Security: Zone: Null
    eth-switch
```

**Meaning** The sample output shows that the Ethernet switching is configured on interface ge-0/0/1 as expected .

**Related Documentation**

- Ethernet Ports Switching Overview for Security Devices on page 104

CHAPTER 13

# Configuring VLANs in Switching Mode



## Understanding VLANs

**Supported Platforms** SRX Series

Each VLAN is a collection of network nodes that are grouped together to form separate broadcast domains. On an Ethernet network that is a single LAN, all traffic is forwarded to all nodes on the LAN. On VLANs, frames whose origin and destination are in the same VLAN are forwarded only within the local VLAN. Frames that are not destined for the local VLAN are the only ones forwarded to other broadcast domains. VLANs thus limit the amount of traffic flowing across the entire LAN, reducing the possible number of collisions and packet retransmissions within a VLAN and on the LAN as a whole.

On an Ethernet LAN, all network nodes must be physically connected to the same network. On VLANs, the physical location of the nodes is not important; therefore, you can group network devices in any way that makes sense for your organization, such as by department or business function, by types of network nodes, or even by physical location. Each VLAN is identified by a single IP subnetwork and by standardized IEEE 802.1Q encapsulation.

To identify which VLAN the traffic belongs to, all frames on an Ethernet VLAN are identified by a tag, as defined in the IEEE 802.1Q standard. These frames are tagged and are encapsulated with 802.1Q tags.

For a simple network that has only a single VLAN, all traffic has the same 802.1Q tag. When an Ethernet LAN is divided into VLANs, each VLAN is identified by a unique 802.1Q tag. The tag is applied to all frames so that the network nodes receiving the frames know to which VLAN a frame belongs. Trunk ports, which multiplex traffic among a number of VLANs, use the tag to determine the origin of frames and where to forward them.

For VLAN configuration details, see Table 14 on page 116.

## Table 14: VLAN Configuration Details

| Field | Function | Action |
|---|---|---|
| **General** | | |
| VLAN Name | Specifies a unique name for the VLAN. | Enter a name.<br>NOTE: VLAN text field is disabled when VLAN tagging is not enabled. |
| VLAN ID/Range | Specifies the identifier or range for the VLAN. | Select one:<br>• **VLAN ID**—Type a unique identification number from **1** through **4094**. If no value is specified, it defaults to 1.<br>• **VLAN Range**—Type a number range to create VLANs with IDs corresponding to the range. For example, the range 2–3 will create two VLANs with the ID 2 and 3. |
| Description | Describes the VLAN. | Enter a brief description for the VLAN. |
| Input Filter | Specifies the VLAN firewall filter that is applied to incoming packets. | To apply an input firewall filter, select the firewall filter from the list. |
| Output Filter | Specifies the VLAN firewall filter that is applied to outgoing packets. | To apply an output firewall filter, select the firewall filter from the list. |
| **Ports** | | |
| Ports | Specifies the ports to be associated with this VLAN for data traffic. You can also remove the port association. | Click one:<br>• **Add**—Select the ports from the available list.<br>• **Remove**—Select the port that you do not want associated with the VLAN. |
| **IP Address** | | |
| Layer 3 Information | Specifies IP address options for the VLAN. | Select to enable the IP address options. |
| IP Address | Specifies the IP address of the VLAN. | Enter the IP address. |
| Subnet Mask | Specifies the range of logical addresses within the address space that is assigned to an organization. | Enter the address, for example, **203.0.113.0**. You can also specify the address prefix. |
| Input Filter | Specifies the VLAN interface firewall filter that is applied to incoming packets. | To apply an input firewall filter to an interface, select the firewall filter from the list. |
| Output Filter | Specifies the VLAN interface firewall filter that is applied to outgoing packets. | To apply an output firewall filter to an interface, select the firewall filter from the list. |
| ARP/MAC Details | Specifies the details for configuring the static IP address and MAC. | Click the **ARP/MAC Details** button. Enter the static IP address and MAC address in the window that is displayed. |
| **VoIP** | | |

Table 14: VLAN Configuration Details *(continued)*

| Field | Function | Action |
|---|---|---|
| Ports | Specifies the ports to be associated with this VLAN for voice traffic. You can also remove the port association. | Click one:<br>• **Add**—Select the ports from the available list.<br>• **Remove**—Select the port that you do not want associated with the VLAN. |

NOTE: On SRX100 devices, dynamic VLAN assignments and guest VLANs are not supported.

On SRX240, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX650 devices, the VLAN range from 1 to 4094 on inet interfaces and the VLAN range from 1 to 3967 on Ethernet switching interfaces. On Ethernet switching interfaces, the VLAN range from 3968 to 4094 falls under the reserved VLAN address range, and the user is not allowed to configure VLANs in this range.

**Related Documentation**

- Example: Configuring VLANs on Security Devices (J-Web Procedure) on page 117
- Ethernet Ports Switching Overview for Security Devices on page 104

## Example: Configuring VLANs on Security Devices (J-Web Procedure)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

This example shows you how to configure a VLAN.

### Requirements

Before you begin:

- Determine which interfaces to use and verify that they are in switching mode. See "Understanding VLANs" on page 115.
- Determine what ports to use on the device and how to segment your network. See "Ethernet Ports Switching Overview for Security Devices" on page 104.

### Overview

In this example, you create a new VLAN and then configure attributes.

### Configuration

**GUI Step-by-Step Procedure**

To access the VLAN:

1. In the J-Web user interface, select **Configure>Switching>VLAN**.

   The VLAN configuration page displays a list of existing VLANs. If you select a specific VLAN, the specific VLAN details are displayed in the details section.

2. Click one:
   - **Add**—Creates a VLAN.
   - **Edit**—Edits an existing VLAN configuration.
   - **Delete**—Deletes an existing VLAN.

   NOTE: If you delete a VLAN, the VLAN configuration for all the associated interfaces is also deleted.

   Add or edit VLAN information.

3. Click one:
   - **OK**—Saves the configuration and returns to the main configuration page, then click **Commit Options**>**Commit**.
   - **Cancel**—Cancels your entries and returns to the main configuration page.

**Related Documentation**
- Understanding VLANs on page 115
- Ethernet Ports Switching Overview for Security Devices on page 104

## Example: Configuring VLANs on Security Devices (CLI Procedure)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

This example shows you how to configure a VLAN.



### Requirements

Before you begin:

- Determine which interfaces to use and verify that they are in switching mode. See "Understanding VLANs" on page 115.
- Determine what ports to use on the device and how to segment your network. See "Ethernet Ports Switching Overview for Security Devices" on page 104.

### Overview

In this example, you create a new VLAN and then configure its attributes. You can configure one or more VLANs to perform Layer 2 switching. The Layer 2 switching functions include

integrated routing and bridging (IRB) for support for Layer 2 switching and Layer 3 IP routing on the same interface. SRX Series devices can function as Layer 2 switches, each with multiple switching or broadcast domains that participate in the same Layer 2 network.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set vlans v10 vlan-id 10
set vlans v10 l3-interface irb.10
set interfaces irb unit 10 family inet address 10.1.1.10/24
set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members 10
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a VLAN:

1. Create the VLAN by setting the unique VLAN name and configuring the VLAN ID.

   ```
   [edit vlans]
   user@host# set vlans v10 vlan-id 10
   ```

2. Bind a Layer 3 interface with the VLAN.

   ```
   [edit]
   user@host# set vlans v10 l3-interface irb.10
   ```

3. Create the subnet for the VLAN's broadcast domain.

   ```
   [edit]
   user@host# set interfaces irb unit 10 family inet address 10.1.1.10/24
   ```

4. Assign an interface to the VLAN by specifying the logical interface (with the unit statement) and specifying the VLAN name as the member.

   ```
   [edit]
   user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members
     10
   ```

**Results**

From configuration mode, confirm your configuration by entering the **show vlans** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show vlans
v10 {
   vlan-id 10;
```

```
    l3-interface irb.10;
}
user@host# show interfaces
ge-0/0/1 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members 10;
            }
        }
    }
}
irb {
    unit 10 {
        family inet {
            address 10.1.1.10/24;
        }
    }
}
```

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

## Verification

### Verifying VLANs

**Purpose** Verify that VLANs are configured and assigned to the interfaces.

**Action** From operational mode, enter the **show vlans** command.

```
user@host> show vlans

Routing instance        VLAN name               Tag          Interfaces
default-switch          default                 1

default-switch          v10                     10
                                                             ge-0/0/1.0
```

**Meaning** The output shows the VLAN is configured and assigned to the interface.

**Related Documentation**

- Understanding VLANs on page 115
- Ethernet Ports Switching Overview for Security Devices on page 104

## Understanding VLAN Retagging on Security Devices

**Supported Platforms** SRX Series, vSRX

VLAN retagging is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D60.

Starting in Junos OS Release 15.1X49-D70, VLAN retagging in switching mode is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Starting in Junos OS Release 15.1X49-D80, VLAN retagging in switching mode is supported on SRX1500 devices.

The VLAN identifier in packets arriving on a Layer 2 trunk port can be rewritten or *retagged* with a different internal VLAN identifier. VLAN retagging is a symmetric operation; upon exiting the same trunk port, the retagged VLAN identifier is replaced with the original VLAN identifier. VLAN retagging provides a way to selectively screen incoming packets and redirect them to a firewall or other security device without affecting other VLAN traffic.

VLAN retagging can be applied only to interfaces configured as Layer 2 trunk interfaces. These interfaces can include redundant Ethernet interfaces in a Layer 2 transparent mode within a chassis cluster configuration.

NOTE: **If a trunk port is configured for VLAN retagging, untagged packets received on the port are not assigned a VLAN identifier with the VLAN retagging configuration. To configure a VLAN identifier for untagged packets received on the physical interface, use the native-vlan-id statement.**

To configure VLAN retagging for a Layer 2 trunk interface, specify a one-to-one mapping of the following:

- Incoming VLAN identifier—VLAN identifier of the incoming packet that is to be retagged. This VLAN identifier must not be the same VLAN identifier configured with the **native-vlan-id** statement for the trunk port.
- Internal VLAN identifier—VLAN identifier for the retagged packet. This VLAN identifier must be in the VLAN identifier list for the trunk port and must not be the same VLAN identifier configured with the **native-vlan-id** statement for the trunk port.

This is an enterprise style of VLAN retagging in which a single command **set interfaces ge-3/0/0 unit 0 family ethernet-switching vlan-rewrite translate 11 2** is sufficient on top of normal trunk configuration. But, in case of Q-in-Q which is is service provider style, the same thing can be done using swap.

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25
- Example: Configuring VLAN Retagging for Layer 2 Transparent Mode on a Security Device on page 33
- Example: Configuring Layer 2 Logical Interfaces on Security Devices on page 6

## Configuring VLAN Retagging on a Layer 2 Trunk Interface of a Security Device

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

VLAN retagging is a feature that works on IEEE standard 802.1Q virtual LAN tagging (VLAN tagging. VLAN retagging for SRX1500 devices is an enterprise style of VLAN retagging, in which a single command is sufficient on top of normal trunk configuration.

1. Create a Layer 2 trunk interface.

```
[edit]
user@host# set interfaces ge-3/0/0 unit 0 family ethernet-switching interface-mode trunk
vlan members 1–10
```

2. Configure VLAN retagging.

```
[edit]
user@host# set interfaces ge-3/0/0 unit 0 family ethernet-switching vlan-rewrite translate
11 2
```

**Related Documentation**

- Understanding VLAN Retagging on Security Devices on page 120

## Example: Configuring a Guest VLAN on a Security Device

**Supported Platforms** SRX300, SRX320, SRX340, SRX345

This example shows how to configure a guest VLAN for limited network access or for Internet-only access to avoid compromising a company's security.

Guest VLANs are not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D60.



### Requirements

Before you begin, verify that the interfaces that will be used are in switch mode. See "Example: Configuring Switching Modes on Security Devices" on page 111 and "Understanding Switching Modes on Security Devices" on page 103.

### Overview

In this example, you configure a VLAN called visitor-vlan with a VLAN ID of 300. Then you set protocols and configure visitor-vlan as the guest VLAN.

### Configuration

**Step-by-Step Procedure**

To configure a guest VLAN:

1. Configure a VLAN.

```
[edit]
```

```
user@host# set vlans visitor-vlan vlan-id 300
```

2. Specify the guest VLAN.

```
[edit]
user@host# set protocols dot1x authenticator interface all guest-vlan visitor-vlan
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

## Verification

To verify the configuration is working properly, enter the **show vlans** and **show protocols dot1x** commands.

**Related Documentation**

- Understanding VLANs on page 115
- Ethernet Ports Switching Overview for Security Devices on page 104

CHAPTER 14

# Configuring Multiple VLAN Registration Protocol



## Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices

**Supported Platforms** SRX300, SRX320, SRX340, SRX345, SRX550M

Multiple VLAN Registration Protocol (MVRP) is a Layer 2 messaging protocol that manages the addition, deletion, and renaming of active virtual LANs, thereby reducing network administrators' time spent on these tasks. Use MVRP on SRX Series devices to dynamically register and unregister active VLANs on trunk interfaces. Using MVRP means that you do not have to manually register VLANs on all connections—that is, you do not need to explicitly bind a VLAN to each trunk interface. With MVRP, you configure a VLAN on one interface and the VLAN configuration is distributed through all active interfaces in the domain.

MVRP is an application protocol of the Multiple Registration Protocol (MRP) and is defined in the IEEE 802.1ak standard. MRP and MVRP replace Generic Attribute Registration Protocol (GARP) and GARP VLAN Registration Protocol (GVRP) and overcome GARP and GVRP limitations.

This topic describes:

## How MVRP Works

When any MVRP-member VLAN is changed, that VLAN sends a protocol data unit (PDU) to all other MVRP-member active VLANs. The PDU informs the other VLANs which devices and interfaces currently belong to the sending VLAN. This way, all MVRP-member VLANs are always updated with the current VLAN state of all other MVRP-member VLANs. Timers dictate when PDUs can be sent and when devices receiving MVRP PDUs can update their MVRP VLAN information.

In addition to sending PDU updates, MVRP dynamically creates VLANs on member interfaces when a new VLAN is added to any one interface. This way, VLANs created on one member device are propagated to other member devices as part of the MVRP message exchange process.

To keep VLAN membership information current, MVRP removes devices and interfaces when they become unavailable. Pruning VLAN information has these benefits:

- Limits the network VLAN configuration to active participants, thereby reducing network overhead.
- Limits broadcast, unknown unicast, and multicast (BUM) traffic to interested devices.

## Basics of MVRP

MVRP is disabled by default on the devices and, when enabled, affects only trunk interfaces. Once you enable MVRP, all VLAN interfaces on the device belong to MVRP (the default **normal** registration mode) and those interfaces accept PDU messages and send their own PDU messages. To prevent one or more interfaces from participating in MVRP, you can specifically configure an interface to **forbidden** registration mode instead of the default **normal** mode.

VLAN updating, dynamic VLAN configuration through MVRP, and VLAN pruning are all active on trunk interfaces when MVRP is enabled.

## MVRP Registration Modes

The MVRP registration mode defines whether an interface does or does not participate in MVRP.

The following MVRP registration modes are configurable:

- forbidden—The interface does not register or declare VLANS (except statically configured VLANs).
- normal—The interface accepts MVRP messages and participates in MVRP. This is the default registration mode setting.
- restricted—The interface ignores all MVRP JOIN messages received for VLANs that are not statically configured on the interface.

## MRP Timers Control MVRP Updates

MVRP registration and updates are controlled by timers that are part of the MRP protocol. These timers are set on a per-interface basis and define when MVRP PDUs can be sent and when MVRP information can be updated on a device.

The following timers are used to control the operation of MVRP:

- Join timer—Controls the interval for the next MVRP PDU transmit opportunity.
- Leave timer—Controls the period of time that an interface on the device waits in the leave state before changing to the unregistered state.
- LeaveAll timer—Controls the frequency with which the interface generates LeaveAll messages.

BEST PRACTICE: **Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.**

## MVRP Uses MRP Messages to Transmit Device and VLAN States

MVRP uses MRP messages to register and declare MVRP states for a device or VLAN and to inform the Layer 2 network that a device or VLAN is leaving MVRP. These messages are communicated as part of the PDU sent by any device interface to the other devices in the network.

The following MRP messages are communicated for MVRP:

- Empty—MVRP information is not declared and no VLAN is registered.
- In—MVRP information is not declared but a VLAN is registered.
- JoinEmpty—MVRP information is declared but no VLAN is registered.
- JoinIn—MVRP information is declared and a VLAN is registered.
- Leave—MVRP information that was previously declared is withdrawn.
- LeaveAll—Unregister all VLANs on the device. VLANs must re-register to participate in MVRP.
- New—The MVRP information is new and a VLAN might not be registered yet.

## MVRP Limitations

The following limitations apply when configuring MVRP:

- MVRP works with Rapid Spanning Tree Protocol (RSTP) and Multiple Spanning Tree Protocol (MSTP), but not with VLAN Spanning Tree Protocol (VSTP).
- MVRP is allowed only on single tagged trunk ports.

- MVRP is not allowed if a physical interface has more than one logical interface.
- MVRP is only allowed if a logical has one trunk interface (unit 0).

**Related Documentation**

- Configuring Multiple VLAN Registration Protocol (MVRP) to Manage Dynamic VLAN Registration on Security Devices on page 128

## Configuring Multiple VLAN Registration Protocol (MVRP) to Manage Dynamic VLAN Registration on Security Devices

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

Starting in Junos OS Release 15.1X49-D80, Multiple VLAN Registration Protocol (MVRP) to manage dynamic VLAN registration is supported on SRX1500 devices. Multiple VLAN Registration Protocol (MVRP) is used to manage dynamic VLAN registration in a Layer 2 network. You can configure MVRP on SRX Series devices.

MVRP is disabled by default on SRX Series devices.

To enable MVRP and to set MVRP options, follow these instructions:

- Enabling MVRP on page 128
- Changing the Registration Mode to Disable Dynamic VLANs on page 128
- Configuring Timer Values on page 129
- Configuring the Multicast MAC Address for MVRP on page 129
- Configuring an MVRP Interface as a Point-to-Point Interface on page 130
- Configuring MVRP Tracing Options on page 130
- Disabling MVRP on page 130

### Enabling MVRP

MVRP can be enabled only on trunk interfaces.

To enable MVRP on a specific trunk interface (here, interface ge-0/0/1):

```
[edit protocols mvrp]
user@host# set interface ge-0/0/1
```

### Changing the Registration Mode to Disable Dynamic VLANs

When the registration mode for an interface is set to **normal** (the default), dynamic VLANs are created on interfaces participating in MVRP. The dynamic VLANs created on one SRX Series device are then propagated by means of MVRP to other SRX Series devices in the topology.

However, dynamic VLAN creation through MVRP can be disabled for all trunk interfaces or for individual trunk interfaces.

## Configuring Timer Values

The timers in MVRP define the amount of time an interface waits to join or leave MVRP or to send or process the MVRP information for the router or switch after receiving an MVRP PDU:

- The join timer controls the amount of time the router or switch waits to accept a registration request.
- The leave timer controls the period of time that the router or switch waits in the Leave state before changing to the unregistered state.
- The leaveall timer controls the frequency with which the LeaveAll messages are communicated.

The default MVRP timer values are 200 ms for the join timer, 1000 ms for the leave timer, and 60 seconds for the leaveall timer.

BEST PRACTICE: **Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.**

To set the join timer at 300 ms for a specific interface (here, interface ge-0/0/1):

```
[edit protocols mvrp]
user@host# set interface ge-0/0/1 join-timer 300
```

To set the leave timer at 400 ms for a specific interface (here, interface ge-0/0/1):

```
[edit protocols mvrp]
user@host# set interface ge-0/0/1 leave-timer 400
```

To set the leaveall timer at 20 seconds for a specific interface (here, interface ge-0/0/1):

```
[edit protocols mvrp]
user@host# set interface ge-0/0/1 leaveall-timer 20
```

**See Also**

- join-timer on page 347
- leave-timer on page 349
- leaveall-timer on page 350

## Configuring the Multicast MAC Address for MVRP

MVRP uses the customer MVRP multicast MAC address when MVRP is enabled. However, you can configure MVRP to use the provider MVRP multicast MAC address instead.

To configure MVRP to use the provider MVRP multicast MAC address:

```
[edit protocols mvrp]
user@host# set bpdu-destination-mac-address provider-bridge-group;
```

See Also
- bpdu-destination-mac-address on page 316

## Configuring an MVRP Interface as a Point-to-Point Interface

Specify that a configured interface is connected point-to-point. If specified, a point-to-point subset of the MRP state machine provides a simpler and more efficient method to accelerate convergence on the network.

To specify that an MVRP interface is point-to-point (here, interface ge-0/0/1):

```
[edit protocols mvrp]
user@host# set interface ge-0/0/1 point-to-point;
```

See Also
- point-to-point on page 359

## Configuring MVRP Tracing Options

Set MVRP protocol-level tracing options.

To specify MVRP protocol tracing (here, the file is **/var/log/mvrp-log**, size is **2m**, number of files is **28**, the option **world-readable** indicates the log can be read by user, and MVRP is flagging **events**):

```
[edit protocols mvrp]
user@host# edit traceoptions file /var/log/mvrp-log size 2m files 28 world-readable flag
  events
```

## Disabling MVRP

MVRP is disabled by default. You need to perform this procedure only if MVRP is previously enabled.

To disable MVRP on all trunk interfaces, use one of the following commands:

```
[edit]
user@host# deactivate protocols mvrp
user@host# delete protocols mvrp
```

See Also
- Understanding VLANs on page 115

CHAPTER 15

# Configuring Q-in-Q Tunneling and VLAN Translation



## Understanding Q-in-Q Tunneling and VLAN Translation on Security Devices

**Supported Platforms** SRX1500, SRX340, SRX345, SRX550M

Q-in-Q tunneling enables service providers on Ethernet access networks to extend a Layer 2 Ethernet connection between two customer sites. Using Q-in-Q tunneling, providers can also segregate or bundle customer traffic into fewer VLANs or different VLANs by adding another layer of 802.1Q tags. Q-in-Q tunneling is useful when customers have overlapping VLAN IDs because customers' VLAN (C-VLAN) tags are prepended by the service-provider VLAN (S-VLAN) tag, which allows you to preserve each customers' VLAN IDs without conflict. The Juniper Networks Junos operating system (Junos OS) implementation of Q-in-Q tunneling supports the IEEE 802.1ad standard.

This topic describes:



### How Q-in-Q Tunneling Works

In Q-in-Q tunneling, as a packet travels from a C-VLAN to an S-VLAN, a service-provider-specific 802.1Q tag is added to the packet. This additional tag is used to segregate traffic into S-VLANs. The original customer 802.1Q tag of the packet is retained and is transmitted transparently, passing through the service provider's network. As the packet leaves the S-VLAN in the downstream direction, the additional 802.1Q tag is removed.

When Q-in-Q tunneling is enabled, trunk interfaces are assumed to be part of the service provider or data center network. Access interfaces are assumed to be customer-facing and accept both tagged and untagged frames. When using many-to-one bundling or mapping a specific interface, you must use the **native** option to specify an S-VLAN for untagged and priority tagged packets if you want to accept these packets. (Priority tagged packets have their VLAN ID set to 0, and their priority code point bits might be configured with a CoS value.) If you do not specify an S-VLAN for them, untagged packets are discarded. The **native** option is not available for all-in-one bundling because there is no need to specify untagged and priority tagged packets when all packets are mapped to an S-VLAN. This topic refers to trunk interfaces as S-VLAN interfaces. This type of interface is also known as a network-to-network interface (NNI). The topic refers to access interfaces as C-VLAN interfaces. This type of interface is also known as a user-network interface (UNI).

Q-in-Q tunneling does not affect any class-of-service (CoS) values that are configured on a C-VLAN. These settings are retained in the C-VLAN tag and can be used after a packet leaves an S-VLAN. CoS values are not copied from C-VLAN tags to S-VLAN tags.

Depending on your interface configuration, you might need to adjust the MTU value on your trunk or access ports to accommodate the 4 bytes used for the tag added by Q-in-Q tunneling. For example, if you use the default MTU value of 1514 bytes on your access and trunk ports, you need to make one of the following adjustments:

- Reduce the MTU on the access links by at least 4 bytes so that the frames do not exceed the MTU of the trunk link when S-VLAN tags are added.
- Increase the MTU on the trunk link so that the link can handle the larger frame size.

NOTE: You can configure Q-in-Q tunneling only on access ports (not trunk ports).

NOTE: You can configure the same interface to be an S-VLAN/NNI interface and a C-VLAN/UNI interface. This means that the same physical interface can transmit single-tagged and double-tagged frames simultaneously. This allows you maximum flexibility in your network topology and lets you maximize the use of your interfaces.

An interface can be a member of multiple S-VLANs. You can map one C-VLAN to one S-VLAN (1:1) or many C-VLANs to many S-VLANs (*N*:*N*). C-VLAN and S-VLAN tags are unique—for instance, you can have both a C-VLAN tag of 101 and an S-VLAN tag of 101. You can limit the set of accepted customer tags to a range of tags or to discrete values. Class-of-service (CoS) values of C-VLANs are unchanged in the downstream direction. You may copy ingress priority and CoS settings to the S-VLAN.

C-VLAN and S-VLAN interfaces accept priority-tagged packets without any configuration.

## How VLAN Translation Works

VLAN translation replaces an incoming C-VLAN tag with an S-VLAN tag instead of adding an additional tag. The C-VLAN tag is therefore lost; because of which a single-tagged packet is normally untagged when it leaves the S-VLAN (at the other end of the link). If an incoming packet has had Q-in-Q tunneling applied in advance, VLAN translation replaces the outer tag and the inner tag is retained when the packet leaves the S-VLAN at the other end of the link.

To configure VLAN translation, use the **mapping swap** statement at the **[edit vlans interface]** hierarchy level.

NOTE: You can configure VLAN translation on access ports only. You cannot configure it on trunk ports, and you cannot configure Q-in-Q tunneling on the same access port.

NOTE: VLAN translation is supported on SRX300 and SRX320 devices and these devices do not support Q-in-Q tunneling.

## Sending and Receiving Untagged Packets

To enable an interface to send and receive untagged packets, you must specify a native VLAN for a physical interface. When the interface receives an untagged packet, it adds the VLAN ID of the native VLAN to the packet and sends the newly tagged packet to the mapped interface.

To specify a native VLAN, use the **native-vlan-id** statement at the **[edit interfaces *interface-name*]** hierarchy level. The native VLAN ID must match the C-VLAN or S-VLAN ID or be included in the VLAN ID list specified on the logical interface.

For example, on a logical interface for a C-VLAN interface, you might specify a C-VLAN ID list of 100-200. Then, on the C-VLAN physical interface, you could specify a native VLAN ID of 150. This configuration would work because the native VLAN of 150 is included in the C-VLAN ID list of 100-200.

We recommend configuring a native VLAN when using any of the approaches to map C-VLANs to S-VLANs. If you do not configure a native VLAN on an interface, untagged packets received by the interface are discarded. See the Mapping C-VLANs to S-VLANs section in this topic for information about the methods of mapping C-VLANs to S-VLANs.

## Disabling MAC Address Learning

In a Q-in-Q deployment, customer packets interfaces are transported without any changes to source and destination MAC addresses. You can disable MAC address learning at the global, interface, and VLAN levels:

- To disable learning globally, disable MAC address learning for the switch.

- To disable learning for an interface, disable MAC address learning for all VLANs of which the specified interface is a member.
- To disable learning for a VLAN, disable MAC address learning for a specified VLAN.

## Mapping C-VLANs to S-VLANs

There are multiple ways to map C-VLANs to S-VLANs:


- Port-based Q-in-Q (All-in-one bundling) on page 134
- Many-to-Many Bundling on page 134
- Mapping a Specific Interface on page 135
- VLAN-Rewrite with Q-in-Q on page 135
- Q-in-Q ethertype on page 135
- Q-in-Q CoS mapping on page 135


If you configure multiple mapping methods, the switch gives priority to mapping a specific interface, then to many-to-many bundling, and last to all-in-one bundling. However, for a particular mapping method, setting up overlapping rules for the same C-VLAN is not supported.

### Port-based Q-in-Q (All-in-one bundling)

All-in-one bundling maps all packets from all C-VLAN interfaces to an S-VLAN.

The C-VLAN interface accepts untagged and single-tagged packets. An S-VLAN 802.1Q tag is then added to these packets, and the packets are sent to the S-VLAN interface, which accepts untagged, single-tagged, and double-tagged packets.

Use the **dot1q-tunneling** statement at the [**edit vlans**] hierarchy to map without specifying customer VLANs. All packets from a specific access interface are mapped to the S-VLAN.

NOTE: The C-VLAN and S-VLAN interfaces accept untagged packets provided that the **native-vlan-id** statement is configured on these interfaces.

### Many-to-Many Bundling

Many-to-many bundling is used to specify which C-VLANs are mapped to which S-VLANs.

Use many-to-many bundling when you want a subset of the C-VLANs on the access switch to be part of multiple S-VLANs. With many-to-many bundling, the C-VLAN interfaces accept untagged and single-tagged packets. An S-VLAN 802.1Q tag is then added to these packets, and the packets are sent to the S-VLAN interfaces, which accept untagged, single-tagged, and double-tagged packets.

NOTE: The C-VLAN and S-VLAN interfaces accept untagged packets provided that the **native-vlan-id** statement is configured on these interfaces.

### Mapping a Specific Interface

Use specific interface mapping when you want to assign an S-VLAN to a specific C-VLAN on an interface. The configuration applies only to the specific interface, not to all access interfaces.

Specific interface mapping has two suboptions: **push** and **swap**. When traffic that is mapped to a specific interface is pushed, the packet retains its original tag as it moves from the C-VLAN to the S-VLAN and an additional S-VLAN tag is added to the packet. When traffic that is mapped to a specific interface is swapped, the incoming tag is replaced with a new VLAN tag. This is sometimes known VLAN rewriting or VLAN translation.

Typically, this method is used to keep data from different customers separate or to provide individualized treatment of the packets on a certain interface. You might also use this method map VLAN traffic from different customers to a single S-VLAN.

When using specific interface mapping, the C-VLAN interfaces accept untagged and single-tagged packets, while the S-VLAN interfaces accept untagged, single-tagged, and double-tagged packets.

NOTE: The C-VLAN and S-VLAN interfaces accept untagged packets provided that the **native-vlan-id** statement is configured on these interfaces.

### VLAN-Rewrite with Q-in-Q

A single UNI interface will encapsulate a customer's CVLAN traffic and classify into a SVLAN, and performs the VLAN translation for other customer's CVLAN and classify into another SVLAN to provide segregation of data traffic received on a UNI interface.

### Q-in-Q ethertype

As per 802.1AD standard, data traffic exiting NNI interface will have SVLAN TPID as 0x88A8; however, 4 SVLAN TPIDs (0x88A8, 0x9100, 0x8100 etc.) are supported based on available ASIC support.

### Q-in-Q CoS mapping

Copy the dot1p bits of CVLAN tagged packets to the dot1p bits of SVLAN tagged packets. This is meant to treat customer VLAN-tagged packets the same way in provider-network.

## Constraints for Q-in-Q Tunneling and VLAN Translation

Be aware of the following constraints when configuring Q-in-Q tunneling and VLAN translation:

- In releases earlier than Junos OS Release 15.1X49-D80, you cannot create a regular VLAN on an interface if you have created an S-VLAN or C-VLAN on that interface for Q-in-Q tunneling. This means that you cannot create an integrated routing and bridging (IRB) interface on that interface because regular VLANs are a required part of IRB configuration. With Junos OS Release 15.1X49-D80, you can create a regular VLAN on a trunk interface that has an S-VLAN, which means that you can also create an IRB interface on the trunk. In this case, the regular VLAN and S-VLAN on the same trunk interface cannot share the same VLAN ID. Junos OS Release 15.1X49-D80 does not support the creation of a regular VLAN on an access interface that has a C-VLAN.
- Most access port security features are not supported with Q-in-Q tunneling and VLAN translation.
- Configuring Q-in-Q tunneling and VLAN rewriting or VLAN translation on the same port is not supported.
- You can configure at most one VLAN rewrite or VLAN translation for a given VLAN and interface. For example, you can create no more than one translation for VLAN 100 on interface xe-0/0/0.
- The combined total of VLANs and rules for Q-in-Q tunneling and VLAN translation cannot exceed 6000. For example, you can configure and commit 4000 VLANs and 2000 rules for Q-in-Q tunneling and VLAN translation. However, you cannot configure 4000 VLANs and 2500 rules for Q-in-Q tunneling and VLAN translation. If you try to commit a configuration that exceeds this limit, you see CLI and system log error messages that alert you the problem.
- MAC addresses are learned from S-VLANs, not C-VLANs.
- Broadcast, unknown unicast, and multicast traffic is forwarded to all members in the S-VLAN.
- The following features are not supported with Q-in-Q tunneling:
  - DHCP relay
  - Fibre Channel over Ethernet
  - IP Source Guard
- The following features are not supported with VLAN rewriting or VLAN translation:
  - Fibre Channel over Ethernet
  - Firewall filter applied to a port or VLAN in the output direction
  - Private VLANs
  - VLAN Spanning Tree Protocol
  - Reflective relay

**Related Documentation**

- Example: Configuring VLANs on Security Devices (CLI Procedure) on page 118

## Configuring Q-in-Q Tunneling on Security Devices

**Supported Platforms** SRX1500, SRX340, SRX345, SRX550M

Q-in-Q tunneling and VLAN translation allow service providers to create a Layer 2 Ethernet connection between two customer sites. Providers can segregate different customers' VLAN traffic on a link (for example, if the customers use overlapping VLAN IDs) or bundle different customer VLANs into a single service VLAN. Data centers can use Q-in-Q tunneling and VLAN translation to isolate customer traffic within a single site or to enable customer traffic flows between cloud data centers in different geographic locations.

NOTE: Q-in-Q VLAN tagging is supported only on SRX340, SRX345, SRX550M, and SRX1500 devices.

NOTE: VLAN translation is supported on SRX300 and SRX320 devices and these devices do not support Q-in-Q tunneling.

Q-in-Q tunneling prepends a service VLAN tag to all customer's 802.1Q VLAN tags. The Juniper Networks Junos OS implementation of Q-in-Q tunneling supports the IEEE 802.1ad standard.

NOTE: This task uses a Junos OS release that supports the Enhanced Layer 2 Software (ELS) configuration style.

With releases earlier than Junos OS Release 15.1X49-D80, you cannot create a regular VLAN on an interface if you have created an S-VLAN or C-VLAN on that interface for Q-in-Q tunneling. This means that you cannot create an integrated routing and bridging (IRB) interface on that interface because regular VLANs are a required part of and IRB configuration. With Junos OS Release 15.1X49-D80, you can create a regular VLAN on a trunk interface that has an S-VLAN, which means that you can also create an IRB interface on the trunk. In this case, the regular VLAN and S-VLAN on the same trunk interface cannot share the same VLAN ID. Junos OS Release 15.1X49-D80, does not allow you to create a regular VLAN on an access interface that has a C-VLAN.

Before setting up Q-in-Q tunneling, make sure you have created and configured the necessary customer VLANs on the neighboring devices. See "Example: Configuring VLANs on Security Devices (J-Web Procedure)" on page 117.



## Using the Different Mapping Methods

Once you have created the required VLANs on the neighboring devices, configure Q-in-Q tunneling using one of the three methods to map customer VLANs (C-VLANs) to service-provider-defined service VLANs (S-VLANs):

- All-in-one bundling maps all packets from all C-VLAN interfaces to an S-VLAN.
- Use many-to-many bundling when you want a subset of the C-VLANs on the access device to be part of multiple S-VLANs.
- Use specific interface mapping when you want to assign an S-VLAN to a specific C-VLAN on an interface.

## Configuring All-in-One Bundling

You can configure Q-in-Q tunneling using the all-in-one bundling method, which forwards all packets entering a C-VLAN interface to an S-VLAN. (Packets are forwarded to the S-VLAN regardless of whether they are tagged or untagged before they enter.) Using this approach saves you the effort of specifying a specific mapping for each C-VLAN.

First configure the S-VLAN and its interface:

1. Enable the interface to transmit packets with two 802.1Q VLAN tags:

   **[edit interfaces *interface-name*]**
   `user@host#` **flexible-vlan-tagging**

2. Enable extended VLAN bridge encapsulation on the interface:

   **[edit interfaces *interface-name*]**
   `user@host#` **encapsulation extended-vlan-bridge**

3. Enable the S-VLAN interface to send and receive untagged packets:

   **[edit interfaces *interface-name*]**
   `user@host#` **native-vlan-id *vlan-id***

4. Bind the logical interface (unit) of the interface to the automatically-created VLAN ID for the S-VLAN:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **vlan-id *number***
   `user@host#` **family ethernet-switching vlan members *vlain-id***

For example, the following configuration enables Q-in-Q tunneling on interface ge-0/0/7, enables ge-0/0/7 to accept untagged packets, and binds the VLAN ID of S-VLAN VL-S91 to a logical interface of ge-0/0/7.

```
set interfaces ge-0/0/7 flexible-vlan-tagging
set interfaces ge-0/0/7 native-vlan-id 91
set interfaces ge-0/0/7 encapsulation extended-vlan-bridge
set interfaces ge-0/0/7 unit 91 vlan-id 91
set interfaces ge-0/0/7 unit 91 family ethernet-switching vlan members VL-S91
```

Now configure all-in-one bundling on a C-VLAN interface:

1. Enable the interface to transmit packets with 802.1Q VLAN tags:

   **[edit interfaces *interface-name*]**
   `user@host#` **flexible-vlan-tagging**

2. Enable extended VLAN bridge encapsulation on the interface:

   **[edit interfaces *interface-name*]**
   `user@host#` **encapsulation extended-vlan-bridge**

3. Enable the C-VLAN interface to send and receive untagged packets:

   **[edit interfaces *interface-name*]**
   `user@host#` **native-vlan-id *vlan-id***

4. Configure a logical interface to receive and forward any tagged packet whose VLAN ID tag matches the list of VLAN IDs you specify:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **vlan-id-list *vlan-id-numbers***

   NOTE: On some SRX Series devices, you can apply no more than eight VLAN identifier lists to a physical interface.

5. Configure the system to add an S-VLAN tag (outer tag) as packets travel from a C-VLAN interface to the S-VLAN:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **input-vlan-map push**

6. Configure the system to remove the S-VLAN tag when packets are forwarded (internally) from the S-VLAN interface to the C-VLAN interface:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **output-vlan-map pop**
   `user@host#` **family ethernet-switching vlan members *vlan-id***

7. Configure S-VLAN and vlan id binding:

   **[edit vlans *vlan-name*]**
   `user@host#` **vlan-id *vlan-id-numbers***

For example, the following configuration makes ge-0/0/4 a member of S-VLAN VL-S91, enables Q-in-Q tunneling, maps packets from C-VLANs to S-VLAN VL-S91, and enables ge-0/0/4 to accept untagged packets. If a packet originates in C-VLAN and needs to be sent across the S-VLAN, a tag with VLAN ID 91 is added to the packet. When a packet is

forwarded (internally) from the S-VLAN interface to interface ge-0/0/4, the tag with VLAN ID 91 is removed.

```
set interfaces ge-0/0/4 flexible-vlan-tagging
set interfaces ge-0/0/4 native-vlan-id 50
set interfaces ge-0/0/4 encapsulation extended-vlan-bridge
set interfaces ge-0/0/4 unit 50 vlan-id-list 30-70
set interfaces ge-0/0/4 unit 50 input-vlan-map push
set interfaces ge-0/0/4 unit 50 output-vlan-map pop
set interfaces ge-0/0/4 unit 50 family ethernet-switching vlan members VL-S91
set vlans VL-S91 vlan-id 91
```

## Configuring Many-to-Many Bundling

You can configure Q-in-Q tunneling using the many-to-many bundling method, which maps packets from multiple C-VLANs to multiple S-VLANs. This method is convenient for mapping a range of C-VLANs without having to specify each one individually. (You can also use this method to configure only one C-VLAN to be mapped to an S-VLAN.)

First configure the S-VLANs and assign them to an interface:

1. Enable the physical interface to transmit packets with two 802.1Q VLAN tags:

   **[edit interfaces *interface-name*]**
   `user@host#` **flexible-vlan-tagging**

2. Enable extended VLAN bridge encapsulation on the interface:

   **[edit interfaces *interface-name*]**
   `user@host#` **encapsulation extended-vlan-bridge**

3. Enable the S-VLAN interface to send and receive untagged packets:

   **[edit interfaces *interface-name*]**
   `user@host#` **native-vlan-id *vlan-id***

4. Bind one of the logical units of the interface to the VLAN ID for one of the S-VLANs.

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **vlan-id *number***
   `user@host#` **family ethernet-switching vlan members *vlan-id***

5. Repeat Step 4 to bind the automatically-created VLAN IDs for the other S-VLANs to the other logical units of the interface:

For example, the following configuration creates S-VLANs VL-S10 and VL-S30 and associates them with interface ge-0/0/7. It also enables Q-in-Q tunneling, enables ge-0/0/7 to accept untagged packets, and maps incoming C-VLAN packets to S-VLANs VL-S10 and VL-S30.

```
set interfaces ge-0/0/7 flexible-vlan-tagging
set interfaces ge-0/0/7 native-vlan-id 10
set interfaces ge-0/0/7 encapsulation extended-vlan-bridge
set interfaces ge-0/0/7 unit 10 vlan-id 10
set interfaces ge-0/0/7 unit 10 family ethernet-switching vlan members VL-S10
```

```
set interfaces ge-0/0/7 unit 30 vlan-id 30
set interfaces ge-0/0/7 unit 30 family ethernet-switching vlan members VL-S30
```

To configure the many-to-many bundling method on a C-VLAN interface, perform the following steps for each customer:

1. Enable the interface to transmit packets with 802.1Q VLAN tags:

   **[edit interfaces *interface-name*]**
   `user@host#` **flexible-vlan-tagging**

2. Enable extended VLAN bridge encapsulation on the interface:

   **[edit interfaces *interface-name*]**
   `user@host#` **encapsulation extended-vlan-bridge**

3. Enable the C-VLAN interface to send and receive untagged packets:

   **[edit interfaces *interface-name*]**
   `user@host#` **native-vlan-id *vlan-id***

4. For each physical interface, configure a logical interface (unit) to receive and forward any tagged packet whose VLAN ID tag matches the list of VLAN IDs you specify:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **vlan-id-list *vlan-id-numbers***

   To configure only one C-VLAN to be mapped to an S-VLAN, specify only one VLAN ID after vlan-id-list.

   NOTE: On some SRX Series devices you can apply no more than eight VLAN identifier list to a physical interface.

5. For each physical interface, configure the system to add an S-VLAN tag (outer tag) as packets travel from the C-VLAN interface to the S-VLAN:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **input-vlan-map push**

6. For each physical interface, configure the system to remove the S-VLAN tag when packets are forwarded from the S-VLAN interface to the C-VLAN interface:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **output-vlan-map pop**
   `user@host#` **family ethernet-switching vlan members *vlan-id***

7. Configure S-VLAN and vlan id binding:

   **[edit vlans *vlan-name*]**
   `user@host#` **vlan-id *vlan-id-numbers***

For example, the following configuration makes ge-0/0/1 a member of S-VLAN VL-S10, enables Q-in-Q tunneling, and maps packets from C-VLANs 10 through 20 to S-VLAN VL-S10. The configuration for customer 2 makes ge-0/0/2 a member of S-VLAN VL-S30, enables Q-in-Q tunneling, and maps packets from C-VLANs 30 through 40, 50 through 60, and 70 through 80 to S-VLAN VL-S30. Both interfaces are configured to accept untagged packets.

If a packet originates in C-VLAN 10 and needs to be sent over the S-VLAN, a tag with a VLAN 10 is added to the packet. If a packet is forwarded internally from the S-VLAN interface to ge-0/0/1, the tag with VLAN 10 is removed. The same principles apply to the C-VLANs configured on interface ge-0/0/2.

NOTE: Notice that you can use the same tag value for an S-VLAN and C-VLAN. For example, the configuration for customer 1 maps C-VLAN 10 to S-VLAN VL-S10. Because C-VLAN and S-VLAN tags use separate name spaces, this configuration is allowed.

Configuration for customer 1:

```
set interfaces ge-0/0/1 flexible-vlan-tagging
set interfaces ge-0/0/1 encapsulation extended-vlan-bridge
set interfaces ge-0/0/1 unit 10 vlan-id-list 10-20
set interfaces ge-0/0/1 native-vlan-id 15
set interfaces ge-0/0/1 unit 10 input-vlan-map push
set interfaces ge-0/0/1 unit 10 output-vlan-map pop
set interfaces ge-0/0/1 unit 10 family ethernet-switching vlan members VL-S10
set vlans VL-S10 vlan-id 10
```

Configuration for customer 2:

```
set interfaces ge-0/0/2 flexible-vlan-tagging
set interfaces ge-0/0/2 encapsulation extended-vlan-bridge
set interfaces ge-0/0/2 unit 30 vlan-id-list 30-40
set interfaces ge-0/0/2 unit 30 vlan-id-list 50-60
set interfaces ge-0/0/2 unit 30 vlan-id-list 70-80
set interfaces ge-0/0/2 native-vlan-id 75
set interfaces ge-0/0/2 unit 30 input-vlan-map push
set interfaces ge-0/0/2 unit 30 output-vlan-map pop
set interfaces ge-0/0/2 unit 30 family ethernet-switching vlan members VL-S30
set vlans VL-S30 vlan-id 30
```

## Configuring a Specific Interface Mapping with VLAN ID Translation Option

You can configure Q-in-Q tunneling by mapping packets from a specified C-VLAN to a specified S-VLAN. In addition, you can configure the system to replace a C-VLAN tag with an S-VLAN tag or replace an S-VLAN tag with a C-VLAN tag (instead of double tagging). This is called VLAN translation or VLAN rewriting. VLAN translation is particularly useful if a service provider's Layer 2 network that connects to customer sites does not support double tagged packets.

When you use VLAN translation, both ends of the link normally must be able to swap the tags appropriately. That is, both ends of the link must be configured to swap the

C-VLAN tag for the S-VLAN tag and swap the S-VLAN tag for the C-VLAN tag so that traffic in both directions is tagged appropriately while in transit and after arrival.

First configure the S-VLAN and its interface:

1. Enable the interface to transmit packets with 802.1Q VLAN tags:

   **[edit interfaces *interface-name*]**
   `user@host#` **flexible-vlan-tagging**

2. Enable the S-VLAN interface to send and receive untagged packets:

   **[edit interfaces *interface-name*]**
   `user@host#` **native-vlan-id *vlan-id***

3. Enable extended VLAN bridge encapsulation on the interface:

   **[edit interfaces *interface-name*]**
   `user@host#` **encapsulation extended-vlan-bridge**

4. Bind the logical interface (unit) of the interface that you specified earlier to the VLAN ID for the S-VLAN:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **vlan-id *number***
   `user@host#` **family ethernet-switching vlan members *vlan-id***

For example, the following configuration enables Q-in-Q tunneling on interface ge-0/0/0, enables ge-0/0/0 to accept untagged packets, and binds a logical interface of ge-0/0/0 to the VLAN ID of S-VLAN VL-S200.

```
set interfaces ge-0/0/0 flexible-vlan-tagging
set interfaces ge-0/0/0 native-vlan-id 10
set interfaces ge-0/0/0 encapsulation extended-vlan-bridge
set interfaces ge-0/0/0 unit 200 vlan-id 200
set interfaces ge-0/0/0 unit 200 family ethernet-switching vlan members VL-S200
```

Now configure a specific interface mapping with optional VLAN ID translation on the C-VLAN interface:

1. Enable the interface to transmit packets with 802.1Q VLAN tags:

   **[edit interfaces *interface-name*]**
   `user@host#` **flexible-vlan-tagging**

2. Enable the C-VLAN interface to send and receive untagged packets:

   **[edit interfaces *interface-name*]**
   `user@host#` **native-vlan-id *vlan-id***

3. Enable extended VLAN bridge encapsulation on the interface:

   **[edit interfaces *interface-name*]**
   `user@host#` **encapsulation extended-vlan-bridge**

4. Configure a logical interface (unit) to receive and forward any tagged packet whose VLAN ID tag matches the VLAN IDs you specify:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **vlan-id *number***

5. Configure the system to remove the existing C-VLAN tag and replace it with the S-VLAN tag when packets enter the C-VLAN interface and are forwarded to the S-VLAN:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **input-vlan-map swap**

6. Configure the system to remove the existing S-VLAN tag and replace it with the C-VLAN tag when packets are forwarded from the S-VLAN interface to the C-VLAN interface:

   **[edit interfaces *interface-name* unit *logical-unit-number*]**
   `user@host#` **output-vlan-map swap**

7. To configure an S-VLAN and associate it with the appropriate C-VLAN interface:

   **[edit vlans *vlan-name*]**
   `user@host#` **interface *interface-name***

8. Configure S-VLAN and vlan id binding:

   **[edit vlans *vlan-name*]**
   `user@host#` **vlan-id *vlan-id-numbers***

For example, the following configuration on C-VLAN interface ge-0/0/1 enables Q-in-Q tunneling, enables ge-0/0/1 to accept untagged packets, and maps incoming packets from C-VLAN 150 to logical interface 200, which is a member of S-VLAN VL-S200. Also, when packets exit from C-VLAN interface ge-0/0/1 and travel to the S-VLAN interface, the C-VLAN tag of 150 is removed and replaced with the S-VLAN tag of 200. When packets travel from the S-VLAN interface to the C-VLAN interface, the S-VLAN tag of 200 is removed and replaced with the C-VLAN tag of 150.

```
set interfaces ge-0/0/1 flexible-vlan-tagging
set interfaces ge-0/0/1 native-vlan-id 10
set interfaces ge-0/0/1 encapsulation extended-vlan-bridge
set interfaces ge-0/0/1 unit 200 vlan-id 150
set interfaces ge-0/0/1 unit 200 family ethernet-switching vlan members VL-S200
set interfaces ge-0/0/1 unit 200 output-vlan-map swap
set interfaces ge-0/0/1 unit 200 input-vlan-map swap
Set vlans VL-S200 vlan-id 200
```

## Configuring VLAN Translation on Security Devices

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

VLAN translation allow service providers to create a Layer 2 Ethernet connection between two customer sites. Providers can segregate different customers' VLAN traffic on a link (for example, if the customers use overlapping VLAN IDs) or bundle different customer VLANs into a single service VLAN. Data centers can use Q-in-Q tunneling to isolate customer traffic within a single site or when customer traffic flows between cloud data centers in different geographic locations.

Before you begin configuring VLAN translation, make sure you have created and configured the necessary customer VLANs on the neighboring switches. See *Configuring VLANs*.

VLAN translation can be done in two ways:

- To configure VLAN translation in VLAN retagging, an enterprise provider style of VLAN translation can be achieved by following CLI configuration:

  **[edit]**
  `user@host#` **set interfaces *intf-name* unit 0 family ethernet-switching interface-mode trunk**
  `user@host#` **set interfaces *intf-name* unit 0 family ethernet-switching vlan members v1000**
  `user@host#` **set interfaces *intf-name* unit 0 family ethernet-switching vlan-rewrite translate 500 1000**

- To configure VLAN translation in Q-in-Q, a service provider style of VLAN translation can be achieved by following CLI configuration:

  **[edit]**
  `user@host#` **set interfaces *intf-name* flexible-vlan-tagging**
  `user@host#` **set interfaces *intf-name* encapsulation extended-vlan-bridge**
  `user@host#` **set interfaces *intf-name* unit 100 vlan-id 500**
  `user@host#` **set interfaces *intf-name* unit 100 input-vlan-map swap**
  `user@host#` **set interfaces *intf-name* unit 100 input-vlan-map tag-protocol-id 0x8100**
  `user@host#` **set interfaces *intf-name* unit 100 output-vlan-map swap**
  `user@host#` **set interfaces *intf-name* unit 100 family ethernet-switching vlan members v1000**

**Related Documentation**

- Understanding Q-in-Q Tunneling and VLAN Translation on Security Devices on page 131

CHAPTER 16

# Configuring Spanning Tree Protocol



## Understanding the Spanning Tree Protocol

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D70, the Spanning Tree Protocol (STP) is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

Spanning Tree Protocol (STP) is not supported from Junos Os Release 15.1X49-D40 to Junos OS Release 15.1X49-D60.

Spanning Tree Protocol (STP), defined in IEEE 802.1D, creates a tree of links in the Ethernet switched network. Links that cause loops in the network are disabled, thereby providing a single active link between any two devices.

Rapid Spanning Tree Protocol (RSTP), originally defined in IEEE 802.1w and later merged into IEEE 802.1D, facilitates faster spanning-tree convergence after a topology change.

Multiple Spanning Tree Protocol (MSTP), initially defined in IEEE 802.1s and later included in IEEE 802.1Q, supports mapping of multiple VLANs onto a single spanning-tree instance.

This reduces the number of spanning-tree instances required in a switched network with many VLANs.

Juniper Networks devices provide Layer 2 loop prevention through STP, RSTP, and MSTP. You can configure bridge protocols data unit (BPDU) protection on interfaces to prevent them from receiving BPDUs that could result in STP misconfigurations, which could lead to network outages.

For STP configuration parameters, see Table 15 on page 148.

### Table 15: STP Configuration Parameters

| Field | Function | Action |
|---|---|---|
| Protocol Name | Displays the spanning-tree protocol. | View only. |
| Disable | Disables STP on the interface. | To enable this option, select the check box. |
| BPDU Protect | Specifies that BPDU blocks are to be processed. | To enable this option, select the check box. |
| Bridge Priority | Specifies the bridge priority. The bridge priority determines which bridge is elected as the root bridge. If two bridges have the same path cost to the root bridge, the bridge priority determines which bridge becomes the designated bridge for a LAN segment. | Select a value. |
| Forward Delay | Specifies the number of seconds an interface waits before changing from spanning-tree learning and listening states to the forwarding state. | Enter a value from 4 through 30 seconds. |
| Hello Time | Specifies time interval in seconds at which the root bridge transmits configuration BPDUs. | Enter a value from 1 through 10 seconds. |
| Max Age | Specifies the maximum aging time in seconds for all MST instances. The maximum aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. | Enter a value from 6 through 40 seconds. |

For RSTP configuration parameters, see Table 16 on page 148.

### Table 16: RSTP Configuration Parameters

| Field | Function | Action |
|---|---|---|
| Protocol Name | Displays the spanning-tree protocol. | View only. |
| Disable | Specifies whether RSTP must be disabled on the interface. | To enable this option, select the check box. |
| BPDU Protect | Specifies that BPDU blocks are to be processed. | To enable this option, select the check box. |

Table 16: RSTP Configuration Parameters *(continued)*

| Field | Function | Action |
|---|---|---|
| Bridge Priority | Specifies the bridge priority. The bridge priority determines which bridge is elected as the root bridge. If two bridges have the same path cost to the root bridge, the bridge priority determines which bridge becomes the designated bridge for a LAN segment. | Select a value. |
| Forward Delay | Specifies the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. | Enter a value from 4 through 30 seconds. |
| Hello Time | Specifies the hello time in seconds for all MST instances. | Enter a value from 1 through 10 seconds. |
| Max Age | Specifies the maximum aging time in seconds for all MST instances. The maximum aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. | Enter a value from 6 through 40 seconds. |

For MSTP configuration parameters, see Table 17 on page 149.

Table 17: MSTP Configuration Parameters

| Field | Function | Action |
|---|---|---|
| Protocol Name | Displays the spanning-tree protocol. | View only. |
| Disable | Specifies whether MSTP must be disabled on the interface. | To enable this option, select the check box. |
| BPDU Protect | Specifies that BPDU blocks are to be processed. | To enable this option, select the check box. |
| Bridge Priority | Specifies the bridge priority. The bridge priority determines which bridge is elected as the root bridge. If two bridges have the same path cost to the root bridge, the bridge priority determines which bridge becomes the designated bridge for a LAN segment. | Select a value. |
| Forward Delay | Specifies the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. | Enter a value from 4 through 30 seconds. |
| Hello Time | Specifies the hello time in seconds for all MST instances. | Enter a value from 1 through 10 seconds. |
| Max Age | Specifies the maximum aging time for all MST instances. The maximum aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. | Enter a value from 6 through 40 seconds. |
| Configuration Name | MSTP region name carried in the MSTP bridge protocol data units (BPDUs). | Enter a name. |

Table 17: MSTP Configuration Parameters *(continued)*

| Field | Function | Action |
|---|---|---|
| Max Hops | Maximum number of hops a BPDU can be forwarded in the MSTP region. | Enter a value from 1 through 255. |
| Revision Level | Revision number of the MSTP region configuration. | Enter a value from 0 through 65,535. |
| **MSTI tab** | | |
| MSTI Id | Specifies the multiple spanning-tree instance (MSTI) identifier. MSTI IDs are local to each region; because of which you can reuse the same MSTI ID in different regions. | Click one:<br>• **Add**—Creates a MSTI.<br>• **Edit**—Edits an existing MSTI.<br>• **Delete**—Deletes an existing MSTI. |
| Bridge Priority | Specifies the bridge priority. The bridge priority determines which bridge is elected as the root bridge. If two bridges have the same path cost to the root bridge, the bridge priority determines which bridge becomes the designated bridge for a LAN segment. | Select a value. |
| VLAN | Specifies the VLANs for the MSTI. | Click one:<br>• **Add**—Selects VLANs from the list.<br>• **Remove**—Deletes the selected VLAN. |
| Interfaces | Specifies the interface for the MSTP protocol. | Click one:<br>• **Add**—Selects interfaces from the list.<br>• **Edit**—Edits the selected interface.<br>• **Remove**—Deletes the selected interface. |

For spanning-tree port configuration details, see Table 18 on page 150.

Table 18: Spanning-Tree Ports Configuration Details

| Field | Function | Action |
|---|---|---|
| Interface Name | Specifies the interface for the spanning-tree protocol type. | Select an interface. |
| Cost | Specifies the link cost to control which bridge is the designated bridge and which interface is the designated interface. | Enter a value from 1 through 200,000,000. |
| Priority | Specifies the interface priority to control which interface is elected as the root port. | Select a value. |
| Edge | Configures the interface as an edge interface. Edge interfaces immediately transition to a forwarding state. | Select to configure the interface as an edge interface. |

Table 18: Spanning-Tree Ports Configuration Details *(continued)*

| Field | Function | Action |
|---|---|---|
| Mode | Specifies the link mode. | Select one:<br>• **Point to Point**—For full-duplex links, select this mode.<br>• **Shared**—For half-duplex links, select this mode. |

**Related Documentation**

- Configuring the Spanning Tree Protocol (J- Web Procedure) on page 151
- Ethernet Ports Switching Overview for Security Devices on page 104
- *Verifying Switching Mode Configuration*

## Configuring the Spanning Tree Protocol (J- Web Procedure)

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D70, the Spanning Tree Protocol (STP) is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

Spanning Tree Protocol (STP) is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D60.

This example shows you how to configure the Spanning Tree Protocol on a Ethernet switched network.



### Requirements

Before you begin:

- Determine which interfaces to use and verify that they are in switch mode. See "Example: Configuring Switching Modes on Security Devices" on page 111.
- Review information about switching modes. See "Understanding Switching Modes on Security Devices" on page 103.

### Overview

In this example, you enable the Spanning Tree Protocol on switched Ethernet ports.

### Configuration

**GUI Step-by-Step Procedure**

To access the Spanning Tree Quick Configuration:

1. In the J-Web user interface, select **Configure>Switching>Spanning Tree**.

The Spanning Tree Configuration page displays a list of existing spanning-trees. If you select a specific spanning tree, the specific spanning tree details are displayed in the General and Interfaces tabs.

2. Click one of the following:
   - **Add**—Creates a spanning tree.
   - **Edit**—Edits an existing spanning-tree configuration.
   - **Delete**—Deletes an existing spanning tree.

   When you are adding a spanning tree, select a protocol name: STP, RSTP, or MSTP.

   Select the **Ports** tab to configure the ports associated with this spanning tree. Click one of the following:
   - **Add**—Creates a new spanning-tree interface configuration.
   - **Edit**—Modifies an existing spanning-tree interface configuration.
   - **Delete**—Deletes an existing spanning-tree interface configuration.

   When you are adding or editing a spanning-tree port, enter information describing the port.

3. Click one:
   - Click **OK** to check your configuration and save it as a candidate configuration, then click **Commit Options**>**Commit**.
   - Click **Cancel** to cancel the configuration without saving changes.

**Related Documentation**
- Understanding the Spanning Tree Protocol on page 147
- Ethernet Ports Switching Overview for Security Devices on page 104
- *Verifying Switching Mode Configuration*

## Configuring the Spanning Tree Protocol (CLI Procedure)

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D70, the Spanning Tree Protocol (STP) is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

Spanning Tree Protocol (STP) is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D60.

This example shows how to configure the Spanning Tree Protocol by using the CLI.

- Requirements on page 153
- Overview on page 153

## Requirements

Before you begin, understand the Spanning Tree Protocol. See "Understanding the Spanning Tree Protocol" on page 147.

## Overview

The default spanning-tree protocol for SRX Series devices is Rapid Spanning Tree Protocol (RSTP). RSTP provides faster convergence times than the original Spanning Tree Protocol (STP). However, some legacy networks require the slower convergence times of basic STP that work with 802.1D 1998 bridges.

If your network includes 802.1D 1998 bridges, you can remove RSTP and explicitly configure STP. When you explicitly configure STP, the devices use the IEEE 802.1D 2004 specification, force version 0. This configuration runs a version of RSTP that is compatible with the classic, basic STP.

This configuration runs a version of RSTP that is compatible with the classic, basic STP.

## Configuration

**CLI Quick Configuration**

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

```
set protocols rstp
set protocols rstp interface ge-0/0/1
set protocols rstp force-version stp interface all
set protocols rstp force-version stp interface ge-0/0/1
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure STP:

1. Configure RSTP on the entire device or on a specific interfaces:

   - To configure RSTP on the entire device:

     ```
     [edit protocols]
     user@host# set rstp
     ```

   - To configure RSTP on a specific interface:

     ```
     [edit protocols]
     user@host# set rstp interface ge-0/0/1
     ```

2. Enable STP either on all interfaces or on a specific interface:

- To enable STP on all interfaces:

```
[edit protocols]
user@host# set rstp force-version stp interface all
```

- To enable STP on a specific interface:

```
[edit protocols]
user@host# set rstp force-version stp interface ge-0/0/1
```

**Results** From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
l2-learning {
    global-mode switching;
}
rstp {
    interface ge-0/0/1;
    interface all;
    force-version stp;
}
```

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

## Verification

### Verifying STP

**Purpose** Verify that STP is configured on your system.

**Action** From operational mode, enter the **show spanning-tree interface** command.

```
user@host> show spanning-tree interface
Spanning tree interface parameters for instance 0

Interface                   Port ID    Designated         Designated         Port
    State  Role
                                         port ID           bridge ID          Cost
ge-0/0/1                      128:2        128:2  32768.307c5e44b250        20000
    BLK    DIS
```

**Meaning** The output shows the STP is configured on an interface ge-0/0/1.

**Related Documentation**

- Understanding the Spanning Tree Protocol on page 147
- Configuring the Spanning Tree Protocol (J- Web Procedure) on page 151

## Understanding BPDU Protection for STP, RSTP, and MSTP

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

Networks frequently use multiple protocols simultaneously to achieve different goals and in some cases those protocols might conflict with each other. One such case is when spanning-tree protocols are active on the network, where a special type of switching frame called a bridge protocol data unit (BPDU) can conflict with BPDUs generated on other devices such as PCs. The different kinds of BPDUs are not compatible, but they can still be recognized by other devices that use BPDUs and cause network outages. You need to protect any device that recognizes BPDUs from picking up incompatible BPDUs.


- Different Kinds of BPDUs on page 155
- Protecting Devices from Incompatible BPDUs on page 155


### Different Kinds of BPDUs

Spanning-tree protocols such as Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP) generate their own BPDUs. These peer STP applications use their BPDUs to communicate, and ultimately, the exchange of BPDUs determines which interfaces block traffic and which interfaces become root ports and forward traffic.

User bridge applications running on a PC can also generate BPDUs. If these BPDUs are picked up by STP applications running on the device, they can trigger STP miscalculations, and those miscalculations can lead to network outages. Similarly, BPDUs generated by STP protocols can cause problems if they are picked up by devices such as PCs that are not using STP. Some mechanism for BPDU protection must be implemented in these cases.

### Protecting Devices from Incompatible BPDUs

To protect the state of spanning-tree protocols on devices from outside BPDUs, enable BPDU protection on the interfaces of a device on which spanning-tree protocols are configured and are connected to user devices (such as PCs)—for example, on edge ports connected to PCs. Use the same strategy when a device on which STP is not configured is connected to a device through a trunk interface that forwards BPDUs generated by spanning-tree protocols. In this case, you protect the device from BPDUs generated by the STP on the device.

To prevent a device from forwarding BPDUs generated by spanning-tree protocols to a device, you can enable **bpdu-block** on an interface.

- On Juniper Networks SRX Series devices that run Juniper Networks Junos operating system (Junos OS) that supports the Enhanced Layer 2 Software (ELS) configuration style, enable **bpdu-block** at the **[edit protocols layer2-control]** hierarchy level. To clear the BPDU error, use **clear error bpdu interface**.

When an interface configured with BPDU protection encounters an incompatible BPDU, it drops that BPDU and then, either shuts down or continues to receive packets other than spanning-tree protocol BPDUs depending on the configuration defined in the

**bpdu-block** statement. If the interface continues to be open after dropping all incompatible BPDUs, all packets except incompatible BPDUs continue to ingress and egress through the interface.

If the interface shuts down after dropping all BPDUs, you can re-enable the interface as follows:

- On Juniper Networks SRX Series devices running Juniper Networks Junos operating system (Junos OS) that supports the Enhanced Layer 2 Software (ELS) configuration style:
  - Include the **disable-timeout** statement at the **[edit protocols layer2-control bpdu-block]** hierarchy level to enable the interfaces to automatically return to service when the specified timer expires.
  - Issue the operational mode command **clear error bpdu interface** on the device.

**Related Documentation**

- Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations on page 156
- Example: Configuring BPDU Protection on Interfaces to Prevent STP Miscalculations on page 160
- Configuring BPDU Protection on Spanning Tree Interfaces on page 165

## Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

SRX Series devices provide Layer 2 loop prevention through Rapid Spanning Tree protocol (RSTP) and Multiple Spanning Tree Protocol (MSTP). All spanning-tree protocols use a special type of frame called a bridge protocol data unit (BPDU) to communicate. Other devices—PC bridging applications, for example, also use BPDUs and generate their own BPDUs. These different BPDUs are not compatible. When BPDUs generated by spanning-tree protocols are transmitted to a device that uses another type of BPDU, they can cause problems on the device. Similarly, if devices within a spanning-tree topology receive BPDUs from other devices, network outages can occur because of STP miscalculations.

This example configures BPDU protection on a SRX Series device that uses RSTP. The upstream configuration is done on the edge interfaces, where outside BPDUs are often received from other devices:

- Requirements on page 157
- Overview on page 157
- Configuration on page 157
- Verification on page 158

## Requirements

This example uses the following software and hardware components:

- Two SRX Series devices in an RSTP topology
- Junos OS Release 15.1X49-D70 or later

Before you configure the interfaces on device 2 for BPDU protection, be sure you have:

- RSTP enabled on the devices.

## Overview

The devices, being in an RSTP, support a loop-free network through the exchange of BPDUs. Receipt of outside BPDUs in an RSTP or MSTP, however, can lead to network outages by triggering an STP misconfiguration. To prevent such outages, enable BPDU protection on spanning tree interfaces that could receive outside BPDUs. If an outside BPDU is received on a BPDU-protected interface, the interface shuts down to prevent the outside BPDU from accessing the spanning tree interface.

In this example, device 1 and device 2 are configured for RSTP. The interfaces on device 2 are edge access ports—edge access ports frequently receive outside BPDUs generated by PC applications.

This example configures interface **ge-0/0/5** and interface **ge-0/0/6** as edge ports on device 2, and then configures BPDU protection on those ports. With BPDU protection enabled, these interfaces shut down when they encounter an outside BPDU sent by the PCs connected to device 2.

## Configuration

To configure BPDU protection on two access interfaces:

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

NOTE: This example configures BPDU protection on specific interfaces. SRX Series devices with support for the Enhanced Layer 2 Software (ELS) configuration style, you can also configure BPDU protection globally on all spanning tree interfaces. See "Configuring BPDU Protection on Spanning Tree Interfaces" on page 165 for additional information.

```
set protocols rstp interface ge-0/0/5 edge
set protocols rstp interface ge-0/0/6 edge
set protocols rstp bpdu-block-on-edge
```

**Step-by-Step Procedure**

To configure RSTP on the two device 2 interfaces, and then configure BPDU protection:

1. Configure RSTP on interface **ge-0/0/5** and interface **ge-0/0/6**, and configure them as edge ports:

```
[edit protocols rstp]
user@host# set interface ge-0/0/5 edge
user@host# set interface ge-0/0/6 edge
```

2. Configure BPDU protection on all edge ports on this device:

```
[edit protocols rstp]
user@host# set bpdu-block-on-edge
```

### Results

Check the results of the configuration:

```
user@host> show configuration protocols rstp
interface ge-0/0/5 {
   edge;
}
interface ge-0/0/6 {
   edge;
}
bpdu-block-on-edge;
```

## Verification

To confirm that the configuration is working properly:

- Displaying the Interface State Before BPDU Protection Is Triggered on page 158
- Verifying That BPDU Protection Is Working Correctly on page 159

### Displaying the Interface State Before BPDU Protection Is Triggered

**Purpose** Before BPDUs can be received from PCs connected to interface **ge-0/0/5** and interface **ge-0/0/6**, confirm the interface state.

**Action** Use the operational mode command:

```
user@host> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
ge-0/0/0     128:513      128:513  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/1     128:514      128:514  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/2     128:515      128:515  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/3     128:516      128:516  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/4     128:517      128:517  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/5     128:518      128:518  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/6     128:519      128:519  32768.0019e2503f00     20000  FWD    DESG
[output truncated]
```

**Meaning** The output from the operational mode command **show spanning-tree interface** shows that **ge-0/0/5** and interface **ge-0/0/6** are ports in a forwarding state.

### Verifying That BPDU Protection Is Working Correctly

**Purpose** In this example, the PCs connected to device 2 start sending BPDUs to interface **ge-0/0/5** and interface **ge-0/0/6**. Verify that BPDU protection is working on the interfaces.

**Action** Use the operational mode command:

```
user@host> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
ge-0/0/0     128:513      128:513  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/1     128:514      128:514  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/2     128:515      128:515  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/3     128:516      128:516  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/4     128:517      128:517  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/5     128:518      128:518  32768.0019e2503f00     20000  BLK    DIS
(Bpdu-Incon)
ge-0/0/6     128:519      128:519  32768.0019e2503f00     20000  BLK    DIS
(Bpdu-Incon)
ge-0/0/7     128:520        128:1  16384.00aabbcc0348     20000  FWD    ROOT
ge-0/0/8     128:521      128:521  32768.0019e2503f00     20000  FWD    DESG
[output truncated]
```

**Meaning** When BPDUs are sent from the PCs to interface **ge-0/0/5** and interface **ge-0/0/6** on device 2, the output from the operational mode command **show spanning-tree interface** shows that the interfaces have transitioned to a BPDU inconsistent state. The BPDU inconsistent state causes the interfaces to shut down.

Disabling the BPDU protection configuration on an interface does not automatically reenable the interface. However, if the **disable-timeout (Spanning Trees)** statement has been included in the BPDU configuration, the interface does return to service after the timer expires. Otherwise, you must use the operational mode command **clear error bpdu** to unblock and reenable the interface.

If the PCs connected to device 2 send BPDUs to the interfaces again, BPDU protection is triggered once more and the interfaces transition back to the BPDU inconsistent state, causing them to shut down. In such cases, you need to find and repair the misconfiguration on the PCs that is sending BPDUs to device 2.

**Related Documentation**

- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Example: Configuring BPDU Protection on Interfaces to Prevent STP Miscalculations on page 160
- Configuring BPDU Protection on Spanning Tree Interfaces on page 165

## Example: Configuring BPDU Protection on Interfaces to Prevent STP Miscalculations

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

NOTE: This example uses Junos OS for SRX Series devices with support for the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

Spanning-tree protocols support loop-free network communication through the exchange of a special type of frame called a bridge protocol data unit (BPDU). However, when BPDUs generated by spanning-tree protocols are communicated to devices on which spanning-tree protocols are not configured, these devices recognize the BPDUs, which can lead to network outages. You can, however, enable BPDU protection on device interfaces to prevent BPDUs generated by spanning-tree protocols from passing through those interfaces. When BPDU protection is enabled, an interface shuts down when any incompatible BPDU is encountered, thereby preventing the BPDUs generated by spanning-tree protocols from reaching the device.

This example configures BPDU protection on STP device downstream interfaces that connect to two PCs:

- Requirements on page 161
- Overview on page 161
- Configuration on page 161
- Verification on page 162

## Requirements

This example uses the following software and hardware components:

- One SRX Series device in an RSTP
- One SRX Series device that is not in any spanning-tree
- Junos OS Release 15.1X49-D70 or later

Before you configure the interfaces on device 2 for BPDU protection, be sure you have:

- Ensured that RSTP is operating on device 1.
- Disabled RSTP on device 2

## Overview

SRX Series devices provide Layer 2 loop prevention through Rapid Spanning Tree protocol (RSTP) and Multiple Spanning Tree Protocol (MSTP). All spanning-tree protocols use a special type of frame called a BPDU to communicate. Other devices also use BPDUs—PC bridging applications, for example, generate their own BPDUs. These different BPDUs are not compatible. When BPDUs generated by spanning-tree protocols are transmitted to a device that uses another type of BPDU, they can cause problems on the device. Similarly, if devices within a spanning-tree topology receive BPDUs from other devices, network outages can occur because of the miscalculations caused by the outside BPDUs. Therefore, you must configure BPDU protection on interfaces in a spanning-tree to avoid network outages.

This example explains how to block outside BPDUs from reaching a device interface connected to devices that are not part of the STP. In this scenario, an interface is shutdown when it encounters an outside BPDU.

This example configures downstream BPDU protection on device 2 interfaces **ge-0/0/5** and **ge-0/0/6**. When BPDU protection is enabled, the device interfaces will shut down if BPDUs generated by the laptops attempt to access device 2.

CAUTION: **When configuring BPDU protection on an interface without spanning trees connected to a device with spanning trees, be careful that you do not configure BPDU protection on all interfaces. Doing so could prevent BPDUs being received on device interfaces (such as a trunk interface) that you intended to have receive BPDUs from a device with spanning trees.**

## Configuration

To configure BPDU protection on the interfaces:

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

NOTE: This example configures BPDU protection on specific interfaces. For, SRX Series devices with support for the Enhanced Layer 2 Software (ELS) configuration style, you can configure BPDU protection globally on all spanning tree interfaces. See "Configuring BPDU Protection on Spanning Tree Interfaces" on page 165 for additional information.

```
set protocols layer2-control bpdu-block interface ge-0/0/5
  set protocols layer2-control bpdu-block interface ge-0/0/6
```

**Step-by-Step Procedure**

To configure BPDU protection for automatic shutdown.

1. To shutdown the BPDU interface on the downstream interface **ge-0/0/5** on device 2:

   ```
   [edit protocol layer 2]
   user@host# set bpdu-block interface ge-0/0/5
   ```

2. To shutdown the BPDU interface on the downstream interface **ge-0/0/6** on device 2:

   ```
   [edit protocol layer 2]
   user@host# set bpdu-block interface ge-0/0/6
   ```

### Results

Check the results of the configuration:

```
user@host> show protocol layer 2
bpdu-block {
  interface ge-0/0/5 {
  interface ge-0/0/6 {
}
```

## Verification

To confirm that the configuration is working properly, perform these tasks:

- Displaying the Interface State Before BPDU Protection Is Triggered on page 162
- Verifying That BPDU Shutdown Protection Is Working Correctly on page 164

### Displaying the Interface State Before BPDU Protection Is Triggered

**Purpose** Before any BPDUs can be received on device 2 on either interface **ge-0/0/5** or interface **ge-0/0/6**, confirm the state of those interfaces.

**Action** Use the operational mode command **show interfaces extensive <interface name>**:

```
user@host> show interfaces extensive ge-0/0/5


Physical interface: ge-0/0/5, Enabled, Physical link is Down
  Interface index: 141, SNMP ifIndex: 516, Generation: 144
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Link-mode: Full-duplex,
Speed: 1000mbps, BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
Source filtering: Disabled,
  Flow control: Disabled, Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 30:7c:5e:44:b1:c6, Hardware address: 30:7c:5e:44:b1:c6
  Last flapped   : 2017-01-16 20:23:55 PST (05:44:46 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
  Dropped traffic statistics due to STP State:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0, FIFO errors: 0,
Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,
 FIFO errors: 0, HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Active alarms  : LINK
  Active defects : LINK
  MAC statistics:                      Receive         Transmit
    Total octets                             0                0
    Total packets                            0                0
    Unicast packets                          0                0
    Broadcast packets                        0                0
    Multicast packets                        0                0
    CRC/Align errors                         0                0
    FIFO errors                              0                0
    MAC control frames                       0                0
    MAC pause frames                         0                0
    Oversized frames                         0
    Jabber frames                            0
    Fragment frames                          0
    VLAN tagged frames                       0
    Code violations                          0
  Filter statistics:
    Input packet count                       0
    Input packet rejects                     0
    Input DA rejects                         0
    Input SA rejects                         0
    Output packet count                                       0
    Output packet pad count                                   0
```

```
    Output packet error count                                0
    CAM destination filters: 1, CAM source filters: 0
  Autonegotiation information:
    Negotiation status: Incomplete
  Packet Forwarding Engine configuration:
    Destination slot: 0
  CoS information:
    Direction : Output
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                                 %           bps     %           usec
    0 best-effort               95     950000000    95              0      low
none
    3 network-control            5      50000000     5              0      low
none
  Interface transmit statistics: Disabled
  MACSec statistics:
    Output
        Secure Channel Transmitted
        Protected Packets               : 0
        Encrypted Packets               : 0
        Protected Bytes                 : 0
        Encrypted Bytes                 : 0
     Input
        Secure Channel Received
        Accepted Packets                : 0
        Validated Bytes                 : 0
        Decrypted Bytes                 : 0
```

**Meaning** The output from the operational mode command **show interfaces extensive** shows that **ge-0/0/5** a is enabled.

### Verifying That BPDU Shutdown Protection Is Working Correctly

**Purpose** Verify that BPDU protection is working correctly in the network by checking to see whether BPDUs have been blocked appropriately.

**Action** Issue **show interfaces extensive <interface name>** to see what happened when the BPDUs reached the two interfaces configured for BPDU protection on device 2:

```
user@host> show interfaces extensive ge-0/0/5
Physical interface: ge-0/0/5, Enabled, Physical link is Down
  Interface index: 659, SNMP ifIndex: 639, Generation: 161
  Link-level type: Ethernet, MTU: 1514, MRU: 0, Link-mode: Auto, Speed: Auto,
  BPDU Error: Detected, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Disabled, Auto-negotiation: Enabled,

  Remote fault: Online, Media type: Copper,
  IEEE 802.3az Energy Efficient Ethernet: Disabled
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 12 supported, 12 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
```

**Meaning** When the BPDUs sent from laptops reached interface **ge-0/0/5** on device 2, the interface transitioned to a BPDU inconsistent state, shutting down the interface to prevent BPDUs from reaching the laptops.

You need to reenable the blocked interface. There are two ways to do this. If you included the statement **disable-timeout(Spanning Trees)** in the BPDU configuration, the interface returns to service after the timer expires. Otherwise, use the operational mode command **clear error bpdu interface *interface-name*** to unblock and reenable **ge-0/0/5**. This command will only reenable an interface but the BPDU configuration for the interface will continue to exist unless you remove the BPDU configuration explicitly.

If BPDUs reach the downstream interface on device 2 again, BPDU protection is triggered again and the interface shuts down. In such cases, you must find and repair the misconfiguration that is sending BPDUs to interface **ge-0/0/5** .

**Related Documentation**

- Configuring BPDU Protection on Spanning Tree Interfaces on page 165
- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations on page 156

## Configuring BPDU Protection on Spanning Tree Interfaces

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

NOTE: This topic applies to Junos OS for SRX Series devices with support for the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

You can configure BPDU protection to ignore BPDU received on interfaces where none is expected. If a BPDU is received on a blocked interface, the interface is disabled and stops forwarding frames. By default, all BPDUs are accepted and processed on all interfaces.

To configure BPDU protection for spanning-tree instance interfaces:

- On a specific spanning-tree interface:
  1. To enable BPDU protection on a specified spanning-tree interface:

     ```
     [edit protocols layer2-control bpdu-block ]
     user@hostt# set interface interface-name
     ```

     If a BPDU is received on the interface, the system will disable the interface and stop forwarding frames out the interface until the bridging process is restarted.

  2. (Optional) Configure the amount of time the system waits before *automatically* unblocking this interface after it has received a BPDU.

     ```
     [edit protocols layer2-contorl bpdu-block interface interface-name]
     user@host# set disable-timeout seconds
     ```

     The range of the *seconds* option value is from 10 through 3600 seconds (one hour). A *seconds* option value of **0** is allowed, but this results in the default behavior (the interface is blocked until the interface is cleared).

- To disable BPDU protection for a specific spanning-tree interface

  ```
  [edit protocols layer2-contorl bpdu-block interface interface-name]
  user@host# set disable-timeout seconds
  ```

**Related Documentation**

- Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations on page 156
- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Configuring BPDU Protection on Spanning Tree Interfaces on page 165

## Understanding Loop Protection for STP, RSTP, and MSTP

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

Juniper Networks SRX Series devices provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP). Loop protection increases the efficiency of STP, RSTP, and MSTP by preventing ports from moving into a forwarding state that would result in a loop opening up in the network.

A loop-free network in spanning-tree topologies is supported through the exchange of a special type of frame called bridge protocol data unit (BPDU). Peer STP applications running on the device interfaces use BPDUs to communicate. Ultimately, the exchange of BPDUs determines which interfaces block traffic (preventing loops) and which interfaces become root ports and forward traffic.

However, a blocking interface can transition to the forwarding state in error if the interface stops receiving BPDUs from its designated port on the segment. Such a transition error can occur when there is a hardware error on the device or software configuration error between the device and its neighbor.

When loop protection is enabled, the spanning-tree topology detects root ports and blocked ports and makes sure both keep receiving BPDUs. If a loop-protection-enabled interface stops receiving BPDUs from its designated port, it reacts as it would react to a problem with the physical connection on this interface. It does not transition the interface to a forwarding state, but instead transitions it to a loop-inconsistent state. The interface recovers and then it transitions back to the spanning-tree blocking state as soon as it receives a BPDU.

We recommend that you enable loop protection on all device interfaces that have a chance of becoming root or designated ports. Loop protection is most effective when enabled in the entire deviced network. When you enable loop protection, you must configure at least one action (**log**, **block**, or both).

Note that an interface can be configured for either loop protection or root protection, but not for both.

**Related Documentation**

- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- **clear error bpdu interface on page 393**
- Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on page 167

## Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

NOTE: This example uses Junos OS for SRX Series devices with support for the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

SRX Series devices provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP). Loop protection increases the efficiency of STP, RSTP, and MSTP by preventing interfaces from moving into a forwarding state that would result in a loop opening up in the network.

This example describes how to configure loop protection for an interface on a SRX Series device in an RSTP topology:



## Requirements

This example uses the following software and hardware components:

- Junos OS Release 15.1X49-D70 or later
- Three SRX Series devices in an RSTP topology

Before you configure the interface for loop protection, be sure you have:

- RSTP operating on the devices.

## Overview

A loop-free network in spanning-tree topologies is supported through the exchange of a special type of frame called bridge protocol data unit (BPDU). Peer STP applications running on the device interfaces use BPDUs to communicate. Ultimately, the exchange of BPDUs determines which interfaces block traffic (preventing loops) and which interfaces become root ports and forward traffic.

A blocking interface can transition to the forwarding state in error if the interface stops receiving BPDUs from its designated port on the segment. Such a transition error can occur when there is a hardware error on the device or software configuration error between the device and its neighbor. When this happens, a loop opens up in the spanning tree. Loops in a Layer 2 topology cause broadcast, unicast, and multicast frames to continuously circle the looped network. As a device processes a flood of frames in a looped network, its resources become depleted and the ultimate result is a network outage.

CAUTION: **An interface can be configured for either loop protection or root protection, but not for both.**

In this example, they are configured for RSTP and create a loop-free topology. Interface **ge-0/0/6** is blocking traffic between device 3 and device 1; thus, traffic is forwarded

through interface **ge-0/0/7** on device 2. BPDUs are being sent from the root bridge on device 1 to both of these interfaces.

This example shows how to configure loop protection on interface **ge-0/0/6** to prevent it from transitioning from a blocking state to a forwarding state and creating a loop in the spanning-tree topology.

A spanning-tree topology contains ports that have specific roles:

- The *root port* is responsible for forwarding data to the root bridge.
- The *alternate port* is a standby port for the root port. When a root port goes down, the alternate port becomes the active root port.
- The *designated port* forwards data to the downstream network segment or device.

This configuration example uses an RSTP topology. However, you also can configure loop protection for MSTP topologies at the [**edit protocols mstp** ] hierarchy level.

## Configuration

To configure loop protection on an interface:

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set protocols rstp interface ge-0/0/6 bpdu-timeout-action block
```

**Step-by-Step Procedure**

To configure loop protection:

1. Configure interface **ge-0/0/6** on device 3:

```
[edit protocols rstp]
user@host# set interface ge-0/0/6 bpdu-timeout-action block
```

### Results

Check the results of the configuration:

```
user@host> show configuration protocols rstp
interface ge-0/0/6 {
  bpdu-timeout-action {
    block;
  }
}
```

## Verification

To confirm that the configuration is working properly, perform these tasks:


- Displaying the Interface State Before Loop Protection Is Triggered on page 170
- Verifying That Loop Protection Is Working on an Interface on page 170


### Displaying the Interface State Before Loop Protection Is Triggered

**Purpose** Before loop protection is triggered on interface **ge-0/0/6**, confirm that the interface is blocking.

**Action** Use the operational mode command:

```
user@host> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
ge-0/0/0     128:513      128:513  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/1     128:514      128:514  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/2     128:515      128:515  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/3     128:516      128:516  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/4     128:517      128:517  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/5     128:518      128:518  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/6     128:519        128:2  16384.00aabbcc0348     20000  BLK    ALT
[output truncated]
```

**Meaning** The output from the operational mode command **show spanning-tree interface** shows that **ge-0/0/6** is the alternate port and in a blocking state.

### Verifying That Loop Protection Is Working on an Interface

**Purpose** Verify the loop protection configuration on interface **ge-0/0/6**. RSTP has been disabled on interface **ge-0/0/4** on device 1. This will stop BPDUs from being sent to interface **ge-0/0/6** and trigger loop protection on the interface.

**Action** Use the operational mode command:

```
user@host> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
ge-0/0/0     128:513      128:513  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/1     128:514      128:514  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/2     128:515      128:515  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/3     128:516      128:516  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/4     128:517      128:517  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/5     128:518      128:518  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/6     128:519      128:519  32768.0019e2503f00     20000  BLK    DIS
(Loop-Incon)
[output truncated]
```

**Meaning** The operational mode command **show spanning-tree interface** shows that interface **ge-0/0/6** has detected that BPDUs are no longer being forwarded to it and has moved into a loop-inconsistent state. The loop-inconsistent state prevents the interface from transitioning to a forwarding state. To clear the BPDU error, issue the operational mode command **clear error bpdu interface** on the device. The interface recovers and transitions back to its original state as soon as it receives BPDUs.

**Related Documentation**

- Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on page 167
- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- clear error bpdu interface on page 393

## Understanding Root Protection for STP, RSTP, and MSTP

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

Juniper Networks SRX Series devices provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP). A loop-free network is supported through the exchange of a special type of frame called bridge protocol data unit (BPDU). Peer STP applications running on the device interfaces use BPDUs to communicate. Ultimately, the exchange of BPDUs determines which interfaces block traffic and which interfaces become root ports and forward traffic.

However, a root port elected through this process has the possibility of being wrongly elected. A user bridge application running on a PC can generate BPDUs, too, and interfere with root port election. Root protection allows network administrators to manually enforce the root bridge placement in the network.

Enable root protection on interfaces that must not receive superior BPDUs from the root bridge and must not be elected as the root port. These interfaces become designated

ports and are typically located on an administrative boundary. If the bridge receives superior STP BPDUs on a port that has root protection enabled, that port transitions to a root-prevented STP state (inconsistency state) and the interface is blocked. This blocking prevents a bridge that must not be the root bridge from being elected the root bridge. After the bridge stops receiving superior STP BPDUs on the interface with root protection, the interface returns to a listening state, followed by a learning state, and ultimately back to a forwarding state. Recovery back to the forwarding state is automatic.

When root protection is enabled on an interface, it is enabled for all the STP instances on that interface. The interface is blocked only for instances for which it receives superior BPDUs. Otherwise, it participates in the spanning-tree topology.

An interface can be configured for either root protection or loop protection, but not for both.

**Related Documentation**

- clear error bpdu interface on page 393
- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on page 167
- Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees on page 172

## Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

NOTE: This example uses Junos OS for SRX Series devices with support for the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

SRX Series devices provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP). Root protection increases the efficiency of STP, RSTP, and MSTP by allowing network administrators to manually enforce the root bridge placement in the network.

This example describes how to configure root protection on an interface on a SRX Series device:

- Requirements on page 173
- Overview on page 173
- Configuration on page 174
- Verification on page 174

## Requirements

This example uses the following software and hardware components:

- Junos OS Release 15.1X49-D70 or later
- Four SRX Series devices in an RSTP topology

Before you configure the interface for root protection, be sure you have:

- RSTP operating on the devices.

## Overview

Peer STP applications running on device interfaces exchange a special type of frame called a bridge protocol data unit (BPDU). Devices communicate interface information using BPDUs to create a loop-free topology that ultimately determines the root bridge and which interfaces block or forward traffic in the spanning tree.

However, a root port elected through this process has the possibility of being wrongly elected. A user bridge application running on a PC can generate BPDUs, too, and interfere with root port election.

To prevent this from happening, enable root protection on interfaces that must not receive superior BPDUs from the root bridge and must not be elected as the root port. These interfaces are typically located on an administrative boundary and are designated ports.

When root protection is enabled on an interface:

- The interface is blocked from becoming the root port.
- Root protection is enabled for all STP instances on that interface.
- The interface is blocked only for instances for which it receives superior BPDUs. Otherwise, it participates in the spanning-tree topology.

CAUTION: **An interface can be configured for either root protection or loop protection, but not for both.**

In this example, they are configured for RSTP and create a loop-free topology. Interface **ge-0/0/7** on device 1 is a designated port on an administrative boundary. It connects to device 4. Device 3 is the root bridge. Interface **ge-0/0/6** on device 1 is the root port.

This example shows how to configure root protection on interface **ge-0/0/7** to prevent it from transitioning to become the root port.

- The *root port* is responsible for forwarding data to the root bridge.
- The *alternate port* is a standby port for the root port. When a root port goes down, the alternate port becomes the active root port.
- The *designated port* forwards data to the downstream network segment or device.

This configuration example uses an RSTP topology. However, you also can configure root protection for STP or MSTP topologies at the [**edit protocols mstp** ] hierarchy level.

## Configuration

To configure root protection on an interface:

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set protocols rstp interface ge-0/0/7 no-root-port
```

**Step-by-Step Procedure**

To configure root protection:

1. Configure interface **ge-0/0/7**:

```
[edit protocols rstp]
user@host#
set interface ge-0/0/7 no-root-port
```

### Results

Check the results of the configuration:

```
user@host> show configuration protocols rstp
interface ge-0/0/7 {
    no-root-port;
}
```

## Verification

To confirm that the configuration is working properly:

- Displaying the Interface State Before Root Protection Is Triggered on page 174
- Verifying That Root Protection Is Working on the Interface on page 175

### Displaying the Interface State Before Root Protection Is Triggered

**Purpose**

Before root protection is triggered on interface **ge-0/0/7**, confirm the interface state.

**Action** Use the operational mode command:

```
user@host> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
ge-0/0/0     128:513      128:513  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/1     128:514      128:514  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/2     128:515      128:515  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/3     128:516      128:516  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/4     128:517      128:517  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/5     128:518        128:2  16384.00aabbcc0348     20000  BLK    ALT
ge-0/0/6     128:519        128:1  16384.00aabbcc0348     20000  FWD    ROOT
ge-0/0/7     128:520      128:520  32768.0019e2503f00     20000  FWD    DESG
[output truncated]
```

**Meaning** The output from the operational mode command **show spanning-tree interface** shows that **ge-0/0/7** is a designated port in a forwarding state.

### Verifying That Root Protection Is Working on the Interface

**Purpose** A configuration change takes place on device 4. A smaller bridge priority on the device 4 causes it to send superior BPDUs to interface **ge-0/0/7**. Receipt of superior BPDUs on interface **ge-0/0/7** will trigger root protection. Verify that root protection is operating on interface **ge-0/0/7**.

**Action** Use the operational mode command:

```
user@host> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
ge-0/0/0     128:513      128:513  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/1     128:514      128:514  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/2     128:515      128:515  32768.0019e2503f00     20000  BLK    DIS
ge-0/0/3     128:516      128:516  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/4     128:517      128:517  32768.0019e2503f00     20000  FWD    DESG
ge-0/0/5     128:518        128:2  16384.00aabbcc0348     20000  BLK    ALT
ge-0/0/6     128:519        128:1  16384.00aabbcc0348     20000  FWD    ROOT
ge-0/0/7     128:520      128:520  32768.0019e2503f00     20000  BLK    DIS
(Root-Incon)
[output truncated]
```

**Meaning** The operational mode command **show spanning-tree interface** shows that interface **ge-0/0/7** has transitioned to a root inconsistent state. The root inconsistent state makes

the interface block, discarding any received BPDUs, and prevents the interface from becoming a candidate for the root port. When the root bridge no longer receives superior STP BPDUs from the interface, the interface will recover and transition back to a forwarding state. Recovery is automatic.

**Related Documentation**

- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Understanding Root Protection for STP, RSTP, and MSTP on page 171
- **disable-timeout (Spanning Trees) on page 319**

CHAPTER 17

# Configuring Link Aggregation Control Protocol



## Understanding Link Aggregation Control Protocol

**Supported Platforms** SRX Series, vSRX

LACP, a subcomponent of IEEE 802.3ad, provides additional functionality for link aggregation groups (LAGs). Use the link aggregation feature to aggregate one or more Ethernet interfaces to form a logical point-to-point link, known as a LAG, virtual link, or bundle. The MAC client can treat this virtual link like a single link.

Starting in Junos OS Release 15.1X49-D80, Link Aggregation Control Protocol (LACP) is supported in Layer 2 transparent mode in addition to existing support in Layer 3 mode for SRX300, SRX320, SRX340, SRX345, SRX1500, SRX4100, SRX4200 devices and vSRX instances. Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, Link Aggregation Control Protocol (LACP) is supported in Layer 2 transparent mode in addition to existing support in Layer 3 mode for SRX5400, SRX5600 and SRX5800 devices. When the SRX Series device uses LACP to bundle the member links, it creates high-speed connections, also known as *fat pipe*, with peer systems. Bandwidth can be increased by adding member links. Increased bandwidth is important especially for redundant Ethernet (reth) and aggregated Ethernet (ae) interfaces, for transmitting and receiving packets to and from the peer end for the whole system. LACP also provides automatic determination, configuration, and monitoring member links. LACP is compatible with other peers that run the 802.3ad LACP protocol. It automatically binds the member links without manually configuring the LAG, thereby avoiding errors.

NOTE: Tentative sessions are created for all interfaces in a particular VLAN. If there is plenty of one-way traffic, numerous tentative sessions are created. When sessions reach the maximum limit, vector fails and packet loss might be seen.

This topic contains the following sections:



## Link Aggregation Benefits

Link aggregation increases bandwidth, provides graceful degradation as failure occurs, and increases availability. It provides network redundancy by load-balancing traffic across all available links. If one of the links fails, the system automatically load-balances traffic across all remaining links.

When LACP is not enabled, a local LAG might attempt to transmit packets to a remote single interface, which causes the communication to fail. When LACP is enabled, a local LAG cannot transmit packets unless a LAG with LACP is also configured on the remote end of the link.

A typical LAG deployment includes aggregate trunk links between an access switch and a distribution switch or customer edge (CE) device.

## Link Aggregation Configuration Guidelines

When configuring link aggregation, note the following guidelines and restrictions:

- Link aggregation is supported only for Ethernet interfaces that are configured in switching mode (**family ethernet-switching**). Aggregating interfaces that are configured in routed mode (**family inet**) is also supported.
- You can configure a LAG by specifying the link number as a physical device and then associating a set of ports with the link. All the ports must have the same speed and be in full-duplex mode. Junos OS assigns a unique ID and port priority to each port. The ID and priority are not configurable.
- You can optionally configure LACP for link negotiation.
- You can optionally configure LACP for link protection.
- You can create up to eight Ethernet ports in each bundle.
- Each LAG must be configured on both sides of the link. The ports on either side of the link must be set to the same speed. At least one end of the LAG must be configured as active.
- LAGs are not supported on virtual chassis port links.
- By default, Ethernet links do not exchange protocol data units (PDUs), which contain information about the state of the link. You can configure Ethernet links to actively transmit PDUs, or you can configure the links to passively transmit them, sending out

LACP PDUs only when they receive them from another link. The transmitting link is known as the actor and the receiving link is known as the partner.

- LAGs can only be used for a point-to-point connection.

For LACP configuration details, see Table 19 on page 179 and Table 20 on page 179.

### Table 19: LACP (Link Aggregation Control Protocol) Configuration

| Field | Function |
|---|---|
| Aggregated Interface | Indicates the name of the aggregated interface. |
| Link Status | Indicates whether the interface is linked (Up) or not linked (Down). |
| VLAN (VLAN ID) | Virtual LAN identifier value for IEEE 802.1Q VLAN tags (0.4094). |
| Description | The description for the LAG. |

### Table 20: Details of Aggregation

| Field | Function |
|---|---|
| Administrative Status | Displays if the interface is enabled (Up) or disabled (Down). |
| Logical Interfaces | Shows the logical interface of the aggregated interface. |
| Member Interfaces | Member interfaces hold all the aggregated interfaces of the selected interfaces. |
| Port Mode | Specifies the mode of operation for the port: trunk or access. |
| Native VLAN (VLAN ID) | VLAN identifier to associate with untagged packets received on the interface. |
| IP Address/Subnet Mask | Specifies the address of the aggregated interfaces. |
| IPV6 Address/Subnet Mask | Specifies the IPV6 address of the aggregated interfaces. |

For aggregated Ethernet interface options, see Table 21 on page 179.

### Table 21: Aggregated Ethernet Interface Options

| Field | Function | Action |
|---|---|---|
| Aggregated Interface | Indicates the name of the aggregated interface. | Enter the aggregated interface name. If an aggregated interface already exists, then the field is displayed as read-only. |

Table 21: Aggregated Ethernet Interface Options *(continued)*

| Field | Function | Action |
|---|---|---|
| LACP Mode | Specifies the mode in which LACP packets are exchanged between the interfaces. The modes are:<br>• None—Indicates that no mode is applicable.<br>• Active—Indicates that the interface initiates transmission of LACP packets<br>• Passive—Indicates that the interface only responds to LACP packets. | Select from the list. |
| Description | The description for the LAG. | Enter the description. |
| Interface | Indicates that the interfaces available for aggregation. | Click **Add** to select the interfaces.<br>NOTE: Only interfaces that are configured with the same speeds can be selected together for a LAG. |
| Speed | Indicates the speed of the interface. | |
| Enable Log | Specifies whether to enable generation of log entries for LAG. | Select to enable log generation. |

NOTE: On SRX100, SRX110, SRX210, SRX220, SRX240, SRX300, SRX320, SRX340, SRX345 and SRX650 devices, the speed mode and link mode configuration are available for member interfaces of ae. (Platform support depends on the Junos OS release in your installation.)

For VLAN options, see Table 22 on page 181.

Table 22: Edit VLAN Options

| Field | Function | Action |
|---|---|---|
| Port Mode | Specifies the mode of operation for the port: trunk or access. | If you select Trunk, you can:<br>1. Click **Add** to add a VLAN member.<br>2. Select the VLAN and click **OK**.<br>3. (Optional) Associate a native VLAN ID with the port.<br>If you select Access, you can:<br>1. Select the VLAN member to be associated with the port.<br>2. (Optional) Associate a VoIP VLAN with the interface. Only a VLAN with a VLAN ID can be associated as a VoIP VLAN.<br>3. Click **OK**. |
| VLAN Options | For trunk interfaces, the VLANs for which the interface can carry traffic. | Click **Add** to select VLAN members. |
| Native VLAN | VLAN identifier to associate with untagged packets received on the interface. | Select the VLAN identifier. |

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D80 | Starting in Junos OS Release 15.1X49-D80, Link Aggregation Control Protocol (LACP) is supported in Layer 2 transparent mode in addition to existing support in Layer 3 mode for SRX300, SRX320, SRX340, SRX345, SRX1500, SRX4100, SRX4200 devices and vSRX instances. |
| 15.1X49-D40 | Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, Link Aggregation Control Protocol (LACP) is supported in Layer 2 transparent mode in addition to existing support in Layer 3 mode for SRX5400, SRX5600 and SRX5800 devices. |

**Related Documentation**

- Example: Configuring Link Aggregation Control Protocol on a Security Device (J-Web Procedure) on page 181
- Ethernet Ports Switching Overview for Security Devices on page 104
- *Verifying Switching Mode Configuration*

## Example: Configuring Link Aggregation Control Protocol on a Security Device (J-Web Procedure)

**Supported Platforms** SRX Series, vSRX

This example shows how to configure LACP.

## Requirements

Before you begin:

- Verify that the Ethernet interfaces are in switch mode. See "Understanding VLANs" on page 115.
- Link aggregation of one or more interfaces must be set up to form a virtual link or link aggregation group (LAG) before you can apply LACP.

## Overview

In this example, you configure link aggregation for switched Ethernet interfaces then apply LACP.

## Configuration

**GUI Step-by-Step Procedure**

To access the LACP Configuration:

1. In the J-Web user interface, select **Configure>Interfaces>Link Aggregation**.

   The Aggregated Interfaces list is displayed.

2. Click one of the following:
   - **Device Count**—Creates an aggregated Ethernet interface, or LAG. You can choose the number of device that you want to create.
   - **Add**—Adds a new aggregated Ethernet Interface, or LAG.
   - **Edit**— Modifies a selected LAG
     - **Aggregation**—Modifies an selected LAG.
     - **VLAN**—Specifies VLAN options for the selected LAG.
     - **IP Option**—Configuring IP address to LAG is not supported and when you try to configure the IP address an error message is displayed.
   - **Delete**—Deletes the selected LAG.
   - **Disable Port** or **Enable Port**—Disables or enables the administrative status on the selected interface.
3. Click one:
   - Click **OK** to check your configuration and save it as a candidate configuration, then click **Commit Options**>**Commit**.
   - Click **Cancel** to cancel the configuration without saving changes.

**Related Documentation**

- Understanding Link Aggregation Control Protocol on page 177
- Ethernet Ports Switching Overview for Security Devices on page 104

# Example: Configuring Link Aggregation Control Protocol on a Security Device (CLI Procedure)

**Supported Platforms** SRX Series, vSRX

This example shows how to configure LACP.



## Requirements

This example uses an SRX Series device.

Before you begin:

- Determine which interfaces to use and verify that they are in switch mode. See "Understanding VLANs" on page 115.

## Overview

In this example, for aggregated Ethernet interfaces, you configure the Link Aggregation Control Protocol (LACP). LACP is one method of bundling several physical interfaces to form one logical interface.

## Configuration

**CLI Quick Configuration**

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

```
set interfaces ge-0/0/6 ether-options 802.3ad ae0
set interfaces ge-0/0/7 ether-options 802.3ad ae0
set interfaces ae0 vlan-tagging
set interfaces ae0 aggregated-ether-options lacp active periodic fast
set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk
set vlan vlan1000 vlan-id 1000
set interfaces ae0 unit 0 family ethernet-switching vlan members vlan1000
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure LACP:

1. Configure the interfaces for ae0.

   [edit ]

```
user@host# set interfaces ge-0/0/6 ether-options 802.3ad ae0
user@host# set interfaces ge-0/0/7 ether-options 802.3ad ae0
```

2. Configure ae0 interface for vlan tagging.

```
[edit ]
user@host# set interfaces ae0 vlan-tagging
```

3. Configure LACP for ae0 and configure periodic transmission of LACP packets.

```
[edit ]
user@host# set interfaces ae0 aggregated-ether-options lacp active periodic fast
```

4. Configure ae0 as a trunk port.

```
[edit ]
user@host# set interfaces ae0 unit 0 family ethernet-switching interface-mode
  trunk
```

5. Configure the VLAN.

```
[edit ]
user@host# set vlan vlan1000 vlan-id 1000
```

6. Add the ae0 interface to the VLAN.

```
[edit ]
user@host# set interfaces ae0 unit 0 family ethernet-switching vlan members
  vlan1000
```

7. If you are done configuring the device, commit the configuration.

```
[edit ]
user@host# commit
```

**Results** From configuration mode, confirm your configuration by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show interfaces
ge-0/0/6 {
  ether-options {
    802.3ad ae0;
  }
}
ge-0/0/7 {
  ether-options {
    802.3ad ae0;
  }
}
ae0 {
```

```
    vlan- tagging;
    aggregated-ether-options {
      lacp {
        active;
        periodic fast;
      }
    }
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members vlan1000;
        }
      }
    }
  }
```

## Verification

### Verifying LACP Statistics

**Purpose** Display LACP statistics for aggregated Ethernet interfaces.

**Action** From operational mode, enter the **show lacp statistics interfaces ae0** command.

```
user@host> show lacp statistics interfaces ae0
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-0/0/6                1352        2035            0            0
      ge-0/0/7                1352        2056            0            0
```

**Meaning** The output shows LACP statistics for each physical interface associated with the aggregated Ethernet interface, such as the following:

- The LACP received counter that increments for each normal hello packet received
- The number of LACP transmit packet errors logged
- The number of unrecognized packet errors logged
- The number of invalid packets received

Use the following command to clear the statistics and see only new changes:

```
user@host# clear lacp statistics interfaces ae0
```

### Verifying LACP Aggregated Ethernet Interfaces

**Purpose** Display LACP status information for aggregated Ethernet interfaces.

**Action** From operational mode, enter the **show lacp interfaces ae0** command.

```
user@host> show lacp interfaces ae0
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      ge-0/0/6       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-0/0/6     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Passive

      ge-0/0/7       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-0/0/7     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Passive

    LACP protocol:        Receive State  Transmit State          Mux State
      ge-0/0/6                  Current   Fast periodic Collecting distributing
      ge-0/0/7                  Current   Fast periodic Collecting distributing
```

**Meaning** The output shows aggregated Ethernet interface information, including the following information:

- The LACP state—Indicates whether the link in the bundle is an actor (local or near-end of the link) or a partner (remote or far-end of the link).
- The LACP mode—Indicates whether both ends of the aggregated Ethernet interface are enabled (active or passive)—at least one end of the bundle must be active.
- The periodic link aggregation control PDU transmit rate.
- The LACP protocol state—Indicates the link is up if it is collecting and distributing packets.

**Related Documentation**

- Understanding Link Aggregation Control Protocol on page 177
- Ethernet Ports Switching Overview for Security Devices on page 104

## Example: Configuring Aggregated Ethernet Device with LAG and LACP (CLI Procedure)

**Supported Platforms** SRX Series, vSRX



### Requirements

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

### Overview

This example shows the configuration of aggregated Ethernet (ae) devices with LAG and LACP.

## Configuration

**Step-by-Step Procedure**

To configure LAG:

1. Configure the number of aggregated Ethernet interfaces with LAG interface that you need to create. Set the device-count option to 5.

   ```
   [edit]
   user@host# set chassis aggregated-devices ethernet device-count 5
   ```

2. Add a port to the aggregated Ethernet interface with LAG.

   ```
   [edit]
   user@host# set interfaces ge-2/0/1 ether-options 802.3ad ae0
   user@host# set interfaces ge-2/0/2 ether-options 802.3ad ae0
   ```

3. Configure LACP for the aggregated Ethernet interface with LAG.

   ```
   [edit]
   user@host# set interfaces ae0 aggregated-ether-options lacp active
   ```

4. Configure family Ethernet switching for the aggregated Ethernet interface with LAG.

   ```
   [edit]
   user@host# set interfaces ae0 unit 0 family ethernet-switching
   ```

5. Configure the VLAN vlan20 with VLAN ID 20.

   ```
   [edit]
   user@host# set vlans vlan20 vlan-id 20
   ```

6. Add the aggregated Ethernet interface to the VLAN.

   ```
   [edit]
   user@host# set vlans vlan20 interface ae0
   ```

7. Check the configuration by entering the **show vlans** and **show interfaces** commands

   ```
   user@host# show vlans
   vlan20 {
       vlan-id 20;
       interface {
           ae0.0;
       }
   }
   ```

   ```
   user@host# show interfaces
   ge-2/0/1 {
       ether-options {
           802.3ad ae0;
       }
   }
   ge-2/0/2 {
       ether-options {
           802.3ad ae0;
   ```

```
    }
}
ae0 {
    aggregated-ether-options {
        lacp {
            active;
        }
    }
    unit 0 {
        family ethernet-switching;
    }
}
```

8. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

NOTE: Likewise, you can configure other devices with LAG and LACP.

## Verification

### Verifying Aggregated Ethernet Interface with LAG and LACP

**Purpose** Verify that you can configure aggregated Ethernet interfaces with LAG and LACP.

**Action** From configuration mode, enter the **show lacp interfaces** to view the LACP interfaces.

```
user@host# run show lacp interfaces
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      ge-2/0/1     Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/0/1   Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/0/2     Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-2/0/2   Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
    LACP protocol:        Receive State  Transmit State          Mux State
      ge-2/0/1                Current   Fast periodic Collecting distributing
      ge-2/0/2                Current   Fast periodic Collecting distributing
```

From configuration mode, enter the **show vlans** command to view the VLAN interfaces.

```
user@host# run show vlans
Name           Tag     Interfaces
default         1      None
vlan20          20     ae0.0
```

From configuration mode, enter the **show interfaces (interface name)** command to view the status of the ge-2/0/1 and ge-2/0/2 interfaces.

```
user@host# run show interfaces ge-2/0/1 terse
Interface               Admin Link Proto    Local                 Remote
```

```
ge-2/0/1                up    up
ge-2/0/1.0              up    up   aenet    --> ae0.0

user@host# run show interfaces ge-2/0/2 terse
Interface               Admin Link Proto    Local                 Remote
ge-2/0/2                up    up
ge-2/0/2.0              up    up   aenet    --> ae0.0
```

**Meaning** The output shows the aggregated Ethernet Interface with LAG and LACP is configured.

**Related Documentation**

- *Understanding Aggregated Ethernet Interfaces*
- *Understanding LACP on Standalone Devices*
- *Example: Configuring Link Aggregation Control Protocol (CLI Procedure)*

CHAPTER 18

# Configuring Class of Service in Switching Mode



## Class of Service Functions in Switching Mode Overview

**Supported Platforms** SRX1500

When a network experiences congestion and delay, some packets must be dropped. Juniper Networks Junos operating system (Junos OS) class of service (CoS) divides traffic into classes to which you can apply different levels of throughput and packet loss when congestion occurs. This allows packet loss to happen according to rules that you configure.

For interfaces that carry IPv4, IPv6, and MPLS traffic, you can configure Junos OS CoS features to provide multiple classes of service for different applications. CoS also allows you to rewrite the Differentiated Services code point (DSCP), IP precedence, 802.1p, or EXP CoS bits of packets egressing an interface, thus allowing you to tailor packets for the remote peers' network requirements.

CoS provides multiple classes of service for different applications. You can configure multiple forwarding classes for transmitting packets, define which packets are placed into each output queue, and schedule the transmission service level for each queue.

In designing CoS applications, you must carefully consider your service needs and thoroughly plan and design your CoS configuration to ensure consistency and interoperability across all platforms in a CoS domain.

**Related Documentation**

- Understanding Junos OS CoS Components for SRX Series Devices on page 192

## Understanding Junos OS CoS Components for SRX Series Devices

**Supported Platforms** SRX1500

This topic describes the Juniper Networks Junos OS class-of-service (CoS) components for Juniper Networks SRX Series devices:



### Code-Point Aliases

A code-point alias assigns a name to a pattern of code-point bits. You can use this name instead of the bit pattern when you configure other CoS components such as classifiers, drop-profile maps, and rewrite rules.

### Policers

Policers limit traffic of a certain class to a specified bandwidth and *burst size*. Packets exceeding the policer limits can be discarded. You define policers with filters that can be associated with input interfaces.

NOTE: You can configure policers to discard packets that exceed the rate limits. If you want to configure CoS parameters such as **loss-priority** and **forwarding-class**, you must use firewall filters.

## Classifiers

Packet classification associates incoming packets with a particular CoS servicing level. In Junos OS, *classifiers* associate packets with a forwarding class and loss priority and assign packets to output queues on the basis of associated forwarding classes. Junos OS supports two general types of classifiers:

- Behavior aggregate or CoS value traffic classifiers—Examines the CoS value in the packet header. The value in this single field determines the CoS settings applied to the packet. BA classifiers allow you to set the forwarding class and loss priority of a packet based on the Differentiated Services code point (DSCP) value, IP precedence value, and IEEE 802.1p value.
- Multifield traffic classifiers—Examines multiple fields in the packet such as source and destination addresses and source and destination port numbers of the packet. With multifield classifiers, you set the forwarding class and loss priority of a packet on the basis of firewall filter rules.

## Forwarding Classes

Forwarding classes group the packets for transmission. Based on forwarding classes, you assign packets to output queues. Forwarding classes affect the forwarding, scheduling, and marking policies applied to packets as they transit a device. By default, four categories of forwarding classes are defined: best effort, assured forwarding, expedited forwarding, and network control. SRX Series devices support, 16 forwarding classes, providing granular classification capability.

## Tail Drop Profiles

Drop profile is a mechanism that defines parameters that enable packets to be dropped from the network. Drop profiles define the meanings of the loss priorities. When you configure drop profiles, you are essentially setting the value for queue fullness. The queue fullness represents a percentage of the queue used to store packets in relation to the total amount that has been allocated for that specific queue.

Loss priorities set the priority of dropping a packet. Loss priority affects the scheduling of a packet without affecting the packet's relative ordering. You can use the loss priority setting to identify packets that have experienced congestion. Typically, you mark packets exceeding some service level with a high loss priority.

## Schedulers

Each switch interface has multiple queues assigned to store packets. The switch determines which queue to service with regard to a particular method of scheduling. This process often involves determining which type of packet must be transmitted before

another. You can define the priority, bandwidth, delay buffer size, and tail drop profiles to be applied to a particular queue for packet transmission.

A scheduler map associates a specified forwarding class with a scheduler configuration. You can associate up to four user-defined scheduler maps with the interfaces.

### Rewrite Rules

A *rewrite rule* sets the appropriate CoS bits in the outgoing packet, thus allowing the next downstream device to classify the packet into the appropriate service group. Rewriting, or marking, outbound packets is useful when the switch is at the border of a network and must alter the CoS values to meet the policies of the targeted peer.

NOTE: Egress firewall filters can also assign forwarding class and loss priority so that the packets are rewritten based on forwarding class and loss priority.

**Related Documentation**

- Class of Service Functions in Switching Mode Overview on page 191

## Classification Overview

**Supported Platforms** SRX Series

*Packet classification* refers to the examination of an incoming packet, which associates the packet with a particular class-of-service (CoS) servicing level. Junos operating system (OS) supports these classifiers:

- Behavior aggregate (BA) classifiers
- Multifield (MF) classifiers
- Default IP precedence classifiers

NOTE: The total number of classifiers supported on a Services Processing Unit (SPU) is 79. Three classifiers are installed on the SPU as default classifiers in the Layer 3 mode, independent of any CoS configuration, which leaves 76 classifiers that can be configured using the CoS CLI commands. The default classifiers number can vary in future releases or in different modes.

Verify the number of default classifiers installed on the SPU to determine how many classifiers can be configured using the CoS CLI commands.

When both BA and MF classifications are performed on a packet, the MF classification has higher precedence.

In Junos OS, classifiers associate incoming packets with a forwarding class (FC) and packet loss priority (PLP), and, based on the associated FC, assign packets to output queues. A packet's FC and PLP specify the behavior of a hop, within the system, to process

the packet. The per-hop behavior (PHB) comprises packet forwarding, policing, scheduling, shaping, and marking. For example, a hop can put a packet in one of the priority queues according to its FC and then manage the queues by checking the packet's PLP. Junos OS supports up to eight FCs and four PLPs.

This topic includes the following sections:



## Behavior Aggregate Classifiers

A BA classifier operates on a packet as it enters the device. Using BA classifiers, the device aggregates different types of traffic into a single FC so that all the types of traffic will receive the same forwarding treatment. The CoS value in the packet header is the single field that determines the CoS settings applied to the packet. BA classifiers allow you to set a packet's FC and PLP based on the Differentiated Services (DiffServ) code point (DSCP) value, DSCP IPv4 value, DSCP IPv6 value, IP precedence value, MPLS EXP bits, or IEEE 802.1p value. The default classifier is based on the IP precedence value. For more information, see "Default IP Precedence Classifier" on page 196.

Junos OS performs BA classification for a packet by examining its Layer 2, Layer 3, and related CoS parameters, as shown in Table 23 on page 195.

### Table 23: BA Classification

| Layer | CoS Parameter |
|---|---|
| Layer 2 | IEEE 802.1p value: User Priority |
| Layer 3 | IPv4 precedence<br>IPv4 Differentiated Services code point (DSCP) value<br>IPv6 DSCP value |

NOTE: **A BA classifier evaluates Layer 2 and Layer 3 parameters independently. The results from Layer 2 parameters override the results from the Layer 3 parameters.**

## Multifield Classifiers

An MF classifier is a second means of classifying traffic flows. Unlike the BA classifier, an MF classifier can examine multiple fields in the packet—for example, the source and destination address of the packet, or the source and destination port numbers of the packet. With MF classifiers, you set the FC and PLP based on firewall filter rules.

NOTE: For a specified interface, you can configure both an MF classifier and a BA classifier without conflicts. Because the classifiers are always applied in sequential order (the BA classifier followed by the MF classifier) any BA classification result is overridden by an MF classifier if they conflict.

Junos OS performs MF traffic classification by directly scrutinizing multiple fields of a packet to classify a packet. This avoids having to rely on the output of the previous BA traffic classification. Junos OS can simultaneously check a packet's data for Layers 2, 3, 4, and 7, as shown in Table 24 on page 196.

Table 24: MF Classification

| Layer | CoS Parameter |
|---|---|
| Layer 2 | IEEE 802.1Q: VLAN ID<br>IEEE 802.1p: User priority |
| Layer 3 | IP precedence value<br>DSCP or DSCP IPv6 value<br>Source IP address<br>Destination IP address<br>Protocol<br>ICMP: Code and type |
| Layer 4 | TCP/UDP: Source port<br>TCP/UDP: Destination port<br>TCP: Flags<br>AH/ESP: SPI |
| Layer 7 | Not supported. |

Using Junos OS, you configure an MF classifier with a firewall filter and its associated match conditions. This enables you to use any filter match criterion to locate packets that require classification.

## Default IP Precedence Classifier

With Junos OS, all logical interface are automatically assigned a default IP precedence classifier when the logical interface is configured. This default traffic classifier maps IP precedence values to an FC and a PLP as shown in Table 25 on page 197. These mapping results are in effect for an ingress packet until the packet is further processed by another classification method.

Table 25: Default IP Precedence Classifier

| IP Precedence CoS Values | Forwarding Class | Packet Loss Priority |
|---|---|---|
| 000 | best-effort | low |
| 001 | best-effort | high |
| 010 | best-effort | low |
| 011 | best-effort | high |
| 100 | best-effort | low |
| 101 | best-effort | high |
| 110 | network-control | low |
| 111 | network-control | high |

**Related Documentation**

- Default Behavior Aggregate Classification on page 198
- Sample Behavior Aggregate Classification on page 199
- *Example: Configuring Behavior Aggregate Classifiers*

## Understanding Packet Loss Priorities

**Supported Platforms** SRX Series, vSRX

Packet loss priorities (PLPs) allow you to set the priority for dropping packets. You can use the PLP setting to identify packets that have experienced congestion. Typically, you mark packets exceeding some service level with a high loss priority—that is, a greater likelihood of being dropped. You set PLP by configuring a classifier or a policer. The PLP is used later in the work flow to select one of the drop profiles used by random early detection (RED).

You can configure the PLP bit as part of a congestion control strategy. The PLP bit can be configured on an interface or in a filter. A packet for which the PLP bit is set has an increased probability of being dropped during congestion.

**Related Documentation**

- Classification Overview on page 194
- Default Behavior Aggregate Classification on page 198
- Sample Behavior Aggregate Classification on page 199
- *Example: Configuring Behavior Aggregate Classifiers*

## Default Behavior Aggregate Classification

**Supported Platforms** SRX Series, vSRX

Table 26 on page 198 shows the forwarding class (FC) and packet loss priority (PLP) that are assigned by default to each well-known Differentiated Services (DiffServ) code point (DSCP). Although several DSCPs map to the expedited-forwarding (ef) and assured-forwarding (af) classes, by default no resources are assigned to these forwarding classes. All af classes other than af1*x* are mapped to best-effort, because RFC 2597, *Assured Forwarding PHB Group*, prohibits a node from aggregating classes. Assignment to the best-effort FC implies that the node does not support that class. You can modify the default settings through configuration.

### Table 26: Default Behavior Aggregate Classification

| DSCP and DSCP IPv6 Alias | Forwarding Class | Packet Loss Priority |
|---|---|---|
| ef | expedited-forwarding | low |
| af11 | assured-forwarding | low |
| af12 | assured-forwarding | high |
| af13 | assured-forwarding | high |
| af21 | best-effort | low |
| af22 | best-effort | low |
| af23 | best-effort | low |
| af31 | best-effort | low |
| af32 | best-effort | low |
| af33 | best-effort | low |
| af41 | best-effort | low |
| af42 | best-effort | low |
| af43 | best-effort | low |
| be | best-effort | low |
| cs1 | best-effort | low |
| cs2 | best-effort | low |
| cs3 | best-effort | low |

Table 26: Default Behavior Aggregate Classification *(continued)*

| DSCP and DSCP IPv6 Alias | Forwarding Class | Packet Loss Priority |
|---|---|---|
| cs4 | best-effort | low |
| cs5 | best-effort | low |
| nc1/cs6 | network-control | low |
| nc2/cs7 | network-control | low |
| other | best-effort | low |

**Related Documentation**

- Classification Overview on page 194
- Sample Behavior Aggregate Classification on page 199
- *Example: Configuring Behavior Aggregate Classifiers*
- Understanding Packet Loss Priorities on page 197

## Sample Behavior Aggregate Classification

**Supported Platforms** SRX Series, vSRX

Table 27 on page 199 shows the device forwarding classes (FCs) associated with each well-known Differentiated Services (DiffServ) code point (DSCP) and the resources assigned to the output queues for a sample DiffServ CoS implementation. This example assigns expedited forwarding to queue 1 and a subset of the assured FCs (af*x*) to queue 2, and distributes resources among all four forwarding classes. Other DiffServ-based implementations are possible.

Table 27: Sample Behavior Aggregate Classification Forwarding Classes and Queues

| DSCP and DSCP IPv6 Alias | DSCP and DSCP IPv6 Bits | Forwarding Class | Packet Loss Priority | Queue |
|---|---|---|---|---|
| ef | 101110 | expedited-forwarding | low | 1 |
| af11 | 001010 | assured-forwarding | low | 2 |
| af12 | 001100 | assured-forwarding | high | 2 |
| af13 | 001110 | assured-forwarding | high | 2 |
| af21 | 010010 | best-effort | low | 0 |
| af22 | 010100 | best-effort | low | 0 |
| af23 | 010110 | best-effort | low | 0 |

Table 27: Sample Behavior Aggregate Classification Forwarding Classes and Queues *(continued)*

| DSCP and DSCP IPv6 Alias | DSCP and DSCP IPv6 Bits | Forwarding Class | Packet Loss Priority | Queue |
|---|---|---|---|---|
| af31 | 011010 | best-effort | low | 0 |
| af32 | 011100 | best-effort | low | 0 |
| af33 | 011110 | best-effort | low | 0 |
| af41 | 100010 | best-effort | low | 0 |
| af42 | 100100 | best-effort | low | 0 |
| af43 | 100110 | best-effort | low | 0 |
| be | 000000 | best-effort | low | 0 |
| cs1 | 0010000 | best-effort | low | 0 |
| cs2 | 010000 | best-effort | low | 0 |
| cs3 | 011000 | best-effort | low | 0 |
| cs4 | 100000 | best-effort | low | 0 |
| cs5 | 101000 | best-effort | low | 0 |
| nc1/cs6 | 110000= | network-control | low | 3 |
| nc2/cs7 | 111000= | network-control | low | 3 |
| other | — | best-effort | low | 0 |

**Related Documentation**

- Classification Overview on page 194
- Default Behavior Aggregate Classification on page 198
- *Example: Configuring Behavior Aggregate Classifiers*
- Understanding Packet Loss Priorities on page 197

## Example: Configuring Behavior Aggregate Classifiers on a Security Device

**Supported Platforms** SRX1500

This example shows how to configure behavior aggregate classifiers for a device to determine forwarding treatment of packets.



## Requirements

Before you begin, determine the forwarding class and PLP that are assigned by default to each well-known DSCP that you want to configure for the behavior aggregate classifier. See "Default Behavior Aggregate Classification" on page 198.

## Overview

You configure behavior aggregate classifiers to classify packets that contain valid DSCPs to appropriate queues. Once configured, you must apply the behavior aggregate classifier to the correct interfaces. You can override the default IP precedence classifier by defining a classifier and applying it to a logical interface. To define new classifiers for all code point types, include the **classifiers** statement at the **[edit class-of-service]** hierarchy level.

In this example, you set the DSCP behavior aggregate classifier to ba-classifier as the default DSCP map. You set a best-effort forwarding class as be-class, an expedited forwarding class as ef-class, an assured forwarding class as af-class, and a network control forwarding class as nc-class. Finally, you apply the behavior aggregate classifier to an IRB interface.

Table 28 on page 201 shows how the behavior aggregate classifier assigns loss priorities, to incoming packets in the four forwarding classes.

**Table 28: Sample ba-classifier Loss Priority Assignments**

| mf-classifier Forwarding Class | For CoS Traffic Type | ba-classifier Assignments |
|---|---|---|
| **be-class** | Best-effort traffic | High-priority code point: **000001** |
| **ef-class** | Expedited forwarding traffic | High-priority code point: **101111** |
| **af-class** | Assured forwarding traffic | High-priority code point: **001100** |
| **nc-class** | Network control traffic | High-priority code point: **110001** |

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service classifiers dscp ba-classifier import default
set class-of-service classifiers dscp ba-classifier forwarding-class be-class loss-priority
  high code-points 000001
set class-of-service classifiers dscp ba-classifier forwarding-class ef-class loss-priority
  high code-points 101111
set class-of-service classifiers dscp ba-classifier forwarding-class af-class loss-priority
  high code-points 001100
set class-of-service classifiers dscp ba-classifier forwarding-class nc-class loss-priority
  high code-points 110001
set class-of-service interfaces irb unit 0 classifiers dscp ba-classifier
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the CLI User Guide.

To configure behavior aggregate classifiers for a device:

1. Configure the class of service.

   ```
   [edit]
   user@host# edit class-of-service
   ```

2. Configure behavior aggregate classifiers for DiffServ CoS.

   ```
   [edit class-of-service]
   user@host# edit classifiers dscp ba-classifier
   user@host# set import default
   ```

3. Configure a best-effort forwarding class classifier.

   ```
   [edit class-of-service classifiers dscp ba-classifier]
   user@host# set forwarding-class be-class loss-priority high code-points 000001
   ```

4. Configure an expedited forwarding class classifier.

   ```
   [edit class-of-service classifiers dscp ba-classifier]
   user@host# set forwarding-class ef-class loss-priority high code-points 101111
   ```

5. Configure an assured forwarding class classifier.

   ```
   [edit class-of-service classifiers dscp ba-classifier]
   user@host# set forwarding-class af-class loss-priority high code-points 001100
   ```

6. Configure a network control forwarding class classifier.

   ```
   [edit class-of-service classifiers dscp ba-classifier]
   user@host# set forwarding-class nc-class loss-priority high code-points 110001
   ```

7. Apply the behavior aggregate classifier to an IRB interface.

   ```
   [edit]
   user@host# set class-of-service interfaces irb unit 0 classifiers dscp ba-classifier
   ```

NOTE: You can use interface wildcards for **interface-name** and **logical-unit-number**.

**Results** From configuration mode, confirm your configuration by entering the **show class-of-service** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show class-of-service
classifiers {
   dscp ba-classifier {
      import default;
      forwarding-class be-class {
         loss-priority high code-points 000001;
      }
      forwarding-class ef-class {
         loss-priority high code-points 101111;
      }
      forwarding-class af-class {
         loss-priority high code-points 001100;
      }
      forwarding-class nc-class {
         loss-priority high code-points 110001;
      }
   }
   forwarding-classes {
      class BE-data queue-num 0;
      class Premium-data queue-num 1;
      class Voice queue-num 2;
      class NC queue-num 3;
   }
   interfaces {
      irb {
         unit 0 {
            classifiers {
               dscp ba-classifier;
            }
         }
      }
      irb {
         unit 0 {
            classifiers {
               dscp v4-ba-classifier;
            }
            irb {
               unit 0 {
                  classifiers {
                     dscp v4-ba-classifier;
                  }
                  irb {
```

```
        unit 0 {
          classifiers {
            dscp v4-ba-classifier;
          }
        }
      }
    }
```

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

## Verification

Confirm that the configuration is working properly.


- Verifying the Code-Point Aliases on page 204
- Verifying the DSCP Classifier on page 205
- Verifying the Forwarding Classes and Output Queues on page 206
- Verifying That the Classifier Is Applied to the Interfaces on page 207


### Verifying the Code-Point Aliases

**Purpose** Make sure that the code-point aliases are configured as expected.

**Action** Run the **show class-of-service code-point-aliases dscp** command.

```
user@host> show class-of-service code-point-aliases dscp

Code point type: dscp
  Alias              Bit pattern
  af11               001010
  af12               001100
  af13               001110
  af21               010010
  af22               010100
  af23               010110
  af31               011010
  af32               011100
  af33               011110
  af41               100010
  af42               100100
  af43               100110
  be                 000000
  be1                000001
  cs1                001000
  cs2                010000
  cs3                011000
  cs4                100000
  cs5                101000
  cs6                110000
  cs7                111000
  ef                 101110
  ef1                101111
  nc1                110000
  nc2                111000
```

**Meaning** The code-point aliases are configured as expected. Note that the custom aliases that you configure are added to the default code-point aliases.

### Verifying the DSCP Classifier

**Purpose** Make sure that the DSCP classifier is configured as expected.

**Action** Run the **show class-of-service classifiers name v4-ba-classifier** command.

```
user@host> show class-of-service classifiers name v4-ba-classifier

Classifier: v4-ba-classifier, Code point type: dscp, Index: 10755
  Code point         Forwarding class                  Loss priority
  000000             BE-data                           high
  000001             BE-data                           low
  000010             BE-data                           low
  000011             BE-data                           low
  000100             BE-data                           low
  000101             BE-data                           low
  000110             BE-data                           low
  000111             BE-data                           low
  001000             BE-data                           low
  001001             BE-data                           low
  001010             Voice                             low
  001011             BE-data                           low
  001100             Voice                             high
  001101             BE-data                           low
  001110             Voice                             high
  001111             BE-data                           low
  010000             BE-data                           low
  010001             BE-data                           low
  010010             BE-data                           low
  010011             BE-data                           low
  010100             BE-data                           low
  010101             BE-data                           low
  010110             BE-data                           low
  010111             BE-data                           low
  011000             BE-data                           low
  011001             BE-data                           low
  011010             BE-data                           low
  011011             BE-data                           low
  011100             BE-data                           low
  011101             BE-data                           low
  011110             BE-data                           low
  011111             BE-data                           low
  100000             BE-data                           low
  100001             BE-data                           low
  100010             BE-data                           low
  100011             BE-data                           low
  100100             BE-data                           low
  100101             BE-data                           low
  100110             BE-data                           low
  100111             BE-data                           low
```

```
  101000              BE-data                         low
  101001              BE-data                         low
  101010              BE-data                         low
  101011              BE-data                         low
  101100              BE-data                         low
  101101              BE-data                         low
  101110              Premium-data                    high
  101111              Premium-data                    low
  110000              NC                              low
  110001              BE-data                         low
  110010              BE-data                         low
  110011              BE-data                         low
  110100              BE-data                         low
  110101              BE-data                         low
  110110              BE-data                         low
  110111              BE-data                         low
  111000              NC                              low
  111001              BE-data                         low
  111010              BE-data                         low
  111011              BE-data                         low
  111100              BE-data                         low
  111101              BE-data                         low
  111110              BE-data                         low
  111111              BE-data                         low
```

**Meaning** Notice that the default classifier is incorporated into the customer classifier. If you were to remove the **import default** statement from the custom classifier, the custom classifier would look like this:

```
user@host> show class-of-service classifier name v4-ba-classifier
Classifier: v4-ba-classifier, Code point type: dscp, Index: 10755
  Code point          Forwarding class                  Loss priority
  000000              BE-data                           high
  000001              BE-data                           low
  101110              Premium-data                      high
  101111              Premium-data                      low
```

### Verifying the Forwarding Classes and Output Queues

**Purpose** Make sure that the forwarding classes are configured as expected.

**Action** Run the **show class-of-service forwarding-class** command.

```
user@host> show class-of-service forwarding-class

Forwarding class                          ID      Queue  Restricted queue  Fabric
priority  Policing priority   SPU priority
  BE-data                                  0       0          0             low
            normal            low
  Premium-data                             1       1          1             low
            normal            low
  Voice                                    2       2          2             low
            normal            low
```

```
  NC                                  3        3          3              low
            normal             low
```

**Meaning** The forwarding classes are configured as expected.

### Verifying That the Classifier Is Applied to the Interfaces

**Purpose** Make sure that the classifier is applied to the correct interfaces.

**Action** Run the **show class-of-service interface** command.

```
user@host> show class-of-service interface irb

Physical interface: irb, Index: 144
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

  Logical interface: irb, Index: 333
Object                 Name                 Type                 Index
Classifier             v4-ba-classifier     dscp                 10755
```

**Meaning** The interfaces are configured as expected.

**Related Documentation**

- *Interfaces Feature Guide for Security Devices*
- Classification Overview on page 194
- Sample Behavior Aggregate Classification on page 199
- Understanding Packet Loss Priorities on page 197

## Example: Configuring and Applying a Firewall Filter for a Multifield Classifier

**Supported Platforms** SRX Series

This example shows how to configure a firewall filter to classify traffic using a multifield classifier. The classifier detects packets of interest to CoS as they arrive on an interface.

- Requirements on page 208
- Overview on page 208
- Configuration on page 208
- Verification on page 211

## Requirements

One common way to detect packets of CoS interest is by source or destination address. The destination address is used in this example, but many other matching criteria for packet detection are available to firewall filters.

## Overview

In this example, you configure the firewall filter mf-classifier. You create and name the assured forwarding traffic class, set the match condition, and specify the destination address as 192.168.44.55. You create the forwarding class for assured forwarding DiffServ traffic as af-class and set the loss priority to low.

Then you create and name the expedited forwarding traffic class, set the match condition, for the expedited forwarding traffic class, and specify the destination address as 192.168.66.77. You then create the forwarding class for expedited forwarding DiffServ traffic as ef-class and set the policer to ef-policer. Then you create and name the network-control traffic class and set the match condition.

You then create and name the forwarding class for the network control traffic class as nc-class. You create and name the forwarding class for the best-effort traffic class as be-class. Finally, you apply the multifield classifier firewall filter as an input filter on each customer-facing or host-facing that needs the filter. In this example, the interface is ge-0/0/0.

## Configuration

**CLI Quick Configuration**

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

```
set firewall filter mf-classifier interface-specific
set firewall filter mf-classifier term assured-forwarding from destination-address
  192.168.44.55
set firewall filter mf-classifier term assured-forwarding then forwarding-class af-class
set firewall filter mf-classifier term assured-forwarding then loss-priority low
set firewall filter mf-classifier term expedited-forwarding from destination-address
  192.168.66.77
set firewall filter mf-classifier term expedited-forwarding then forwarding-class ef-class
set firewall filter mf-classifier term expedited-forwarding then policer ef-policer
set firewall filter mf-classifier term network-control from precedence net-control
set firewall filter mf-classifier term network-control then forwarding-class nc-class
set firewall filter mf-classifier term best-effort then forwarding-class be-class
set interfaces irb unit 0 family inet filter input mf-classifier
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a firewall filter for a multifield classifier for a device:

1. Create and name the multifield classifier filter.

```
[edit]
user@host# edit firewall filter mf-classifier
user@host# set interface-specific
```

2. Create and name the term for the assured forwarding traffic class.

```
[edit firewall filter mf-classifier]
user@host# edit term assured-forwarding
```

3. Specify the destination address for assured forwarding traffic.

```
[edit firewall filter mf-classifier term assured-forwarding]
user@host# set from destination-address 192.168.44.55
```

4. Create the forwarding class and set the loss priority for the assured forwarding traffic class.

```
[edit firewall filter mf-classifier term assured-forwarding]
user@host# set then forwarding-class af-class
user@host# set then loss-priority low
```

5. Create and name the term for the expedited forwarding traffic class.

```
[edit]
user@host# edit firewall filter mf-classifier
user@host# edit term expedited-forwarding
```

6. Specify the destination address for the expedited forwarding traffic.

```
[edit firewall filter mf-classifier term expedited-forwarding]
user@host# set from destination-address 192.168.66.77
```

7. Create the forwarding class and apply the policer for the expedited forwarding traffic class.

```
[edit firewall filter mf-classifier term expedited-forwarding]
user@host# set then forwarding-class ef-class
user@host# set then policer ef-policer
```

8. Create and name the term for the network control traffic class.

```
[edit]
user@host# edit firewall filter mf-classifier
user@host# edit term network-control
```

9. Create the match condition for the network control traffic class.

```
[edit firewall filter mf-classifier term network-control]
user@host# set from precedence net-control
```

10. Create and name the forwarding class for the network control traffic class.

```
[edit firewall filter mf-classifier term network-control]
user@host# set then forwarding-class nc-class
```

11. Create and name the term for the best-effort traffic class.

```
[edit]
user@host# edit firewall filter mf-classifier
user@host# edit term best-effort
```

12. Create and name the forwarding class for the best-effort traffic class.

```
[edit firewall filter mf-classifier term best-effort]
user@host# set then forwarding-class be-class
```

NOTE: Because this is the last term in the filter, it has no match condition.

13. Apply the multifield classifier firewall filter as an input filter.

```
[edit]
user@host# set interfaces irb unit 0 family inet filter input mf-classifier
```

**Results** From configuration mode, confirm your configuration by entering the **show firewall filter mf-classifier** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host# show firewall filter mf-classifier
  interface-specific;
  term assured-forwarding {
  from {
    destination-address {
    192.168.44.55;
  }
  }
  then {
    loss-priority low;
    forwarding-class af-class;
  }
  }
  term expedited-forwarding {
    from {
    destination-address {
```

```
    192.168.66.77;
  }
  }
  then {
    policer ef-policer;
    forwarding-class ef-class;
  }
  }
  term network-control {
    from {
  precedence net-control;
  }
  then forwarding-class nc-class;
    }
    term best-effort {
  then forwarding-class be-class;
}
```

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

## Verification

### Verifying a Firewall Filter for a Multifield Classifier Configuration

**Purpose** Verify that a firewall filter for a multifield classifier is configured properly on a device.

**Action** From configuration mode, enter the **show firewall filter mf-classifier** command.

**Related Documentation**

- Understanding Junos OS CoS Components for SRX Series Devices on page 192

# Rewrite Rules Overview

**Supported Platforms** SRX Series, vSRX

A rewrite rule modifies the appropriate class-of-service (CoS) bits in an outgoing packet. Modification of CoS bits allows the next downstream device to classify the packet into the appropriate service group. Rewriting or marking outbound packets is useful when the device is at the border of a network and must alter the CoS values to meet the policies of the targeted peer. A rewrite rule examines the forwarding class and loss priority of a packet and sets its bits to a corresponding value specified in the rule.

Typically, a device rewrites CoS values in outgoing packets on the outbound interfaces of an edge device, to meet the policies of the targeted peer. After reading the current forwarding class and loss priority information associated with the packet, the transmitting device locates the chosen CoS value from a table, and writes this CoS value into the packet header.

NOTE: You can configure up to 32 IEEE 802.1p rewrite rules on each SRX5K-MPC on the SRX5600 and SRX5800 devices.

**Related Documentation**

- Example: Configuring and Applying Rewrite Rules on a Security Device on page 213

## Rewriting Frame Relay Headers

**Supported Platforms** SRX Series



### Assigning the Default Frame Relay Rewrite Rule to an Interface

For Juniper Networks device interfaces with Frame Relay encapsulation, you can rewrite the discard eligibility (DE) bit based on the loss priority of Frame Relay traffic. For each outgoing frame with the loss priority set to low, medium-low, medium-high, or high, you can set the DE bit CoS value to **0** or **1**. You can combine a Frame Relay rewrite rule with other rewrite rules on the same interface. For example, you can rewrite both the DE bit and MPLS EXP bit.

The default Frame Relay rewrite rule contains the following settings:

```
loss-priority low code-point 0;
loss-priority medium-low code-point 0;
loss-priority medium-high code-point 1;
loss-priority high code-point 1;
```

This default rule sets the DE CoS value to **0** for each outgoing frame with the loss priority set to low or medium-low. This default rule sets the DE CoS value to **1** for each outgoing frame with the loss priority set to medium-high or high.

To assign the default rule to an interface, include the **frame-relay-de default** statement at the **[edit class-of-service interfaces interface *interface-name* unit *logical-unit-number* unit rewrite-rules]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]
frame-relay-de default;
```

### Defining a Custom Frame Relay Rewrite Rule

To define a custom Frame Relay rewrite rule, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
rewrite-rules {
   frame-relay-de rewrite-name {
      import (rewrite-name | default);
      forwarding-class class-name {
         loss-priority level code-point (0 | 1);
```

```
    }
  }
}
```

A custom rewrite rule sets the DE bit to the **0** or **1** CoS value based on the assigned loss priority of low, medium-low, medium-high, or high for each outgoing frame.

The rule does not take effect until you apply it to a logical interface. To apply the rule to a logical interface, include the **frame-relay-de *map-name*** statement at the **[edit class-of-service interfaces interface *interface-name* unit *logical-unit-number* rewrite-rules]** hierarchy level:

[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]
frame-relay-de *map-name*;

**Related Documentation**

- Rewrite Rules Overview on page 211
- Example: Configuring and Applying Rewrite Rules on a Security Device on page 213

## Example: Configuring and Applying Rewrite Rules on a Security Device

**Supported Platforms** SRX Series, vSRX

This example shows how to configure and apply rewrite rules for a device.



### Requirements

Before you begin, create and configure the forwarding classes.

### Overview

You can configure rewrite rules to replace CoS values on packets received from the customer or host with the values expected by other devices. You do not have to configure rewrite rules if the received packets already contain valid CoS values. Rewrite rules apply the forwarding class information and packet loss priority used internally by the device to establish the CoS value on outbound packets. After you configure rewrite rules, you must apply them to the correct interfaces.

In this example, you configure the rewrite rule for DiffServ CoS as rewrite-dscps. You specify the best-effort forwarding class as be-class, expedited forwarding class as ef-class, an assured forwarding class as af-class, and a network control class as nc-class. Finally, you apply the rewrite rule to an IRB interface.

NOTE: **You can apply one rewrite rule to each logical interface.**

Table 29 on page 214 shows how the rewrite rules replace the DSCPs on packets in the four forwarding classes.

Table 29: Sample rewrite-dscps Rewrite Rules to Replace DSCPs

| mf-classifier Forwarding Class | For CoS Traffic Type | rewrite-dscps Rewrite Rules |
|---|---|---|
| be-class | Best-effort traffic—Provides no special CoS handling of packets. Typically, RED drop profile is aggressive and no loss priority is defined. | Low-priority code point: 000000<br>High-priority code point: 000001 |
| ef-class | Expedited forwarding traffic—Provides low loss, low delay, low jitter, assured bandwidth, and end-to-end service. Packets can be forwarded out of sequence or dropped. | Low-priority code point: 101110<br>High-priority code point: 101111 |
| af-class | Assured forwarding traffic—Provides high assurance for packets within the specified service profile. Excess packets are dropped. | Low-priority code point: 001010<br>High-priority code point: 001100 |
| nc-class | Network control traffic—Packets can be delayed, but not dropped. | Low-priority code point: 110000<br>High-priority code point: 110001 |

NOTE: Forwarding classes can be configured in a DSCP rewriter and also as an action of an IDP policy to rewrite DSCP code points. To ensure that the forwarding class is used as an action in an IDP policy, it is important that you do not configure an IDP policy and interface-based rewrite rules with the same forwarding class.

## Configuration

- [xref target has no title]

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class be-class
    loss-priority low code-point 000000
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class be-class
    loss-priority high code-point 000001
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class ef-class loss-priority
    low code-point 101110
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class ef-class loss-priority
    high code-point 101111
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class af-class loss-priority
    low code-point 001010
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class af-class loss-priority
    high code-point 001100
```

```
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class nc-class loss-priority
  low code-point 110000
set class-of-service rewrite-rules dscp rewrite-dscps forwarding-class nc-class loss-priority
  high code-point 110001
set class-of-service interfaces irb unit 0 rewrite-rules dscp rewrite-dscps
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure and apply rewrite rules for a device:

1. Configure rewrite rules for DiffServ CoS.

```
[edit]
user@host# edit class-of-service
user@host# edit rewrite-rules dscp rewrite-dscps
```

2. Configure best-effort forwarding class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class be-class loss-priority low code-point 000000
user@host# set forwarding-class be-class loss-priority high code-point 000001
```

3. Configure expedited forwarding class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class ef-class loss-priority low code-point 101110
user@host# set forwarding-class ef-class loss-priority high code-point 101111
```

4. Configure an assured forwarding class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class af-class loss-priority low code-point 001010
user@host# set forwarding-class af-class loss-priority high code-point 001100
```

5. Configure a network control class rewrite rules.

```
[edit class-of-service rewrite-rules dscp rewrite-dscps]
user@host# set forwarding-class nc-class loss-priority low code-point 110000
user@host# set forwarding-class nc-class loss-priority high code-point 110001
```

6. Apply rewrite rules to an IRB interface.

```
[edit class-of-service]
user@host# set interfaces irb unit 0 rewrite-rules dscp rewrite-dscps
```

**Results**

From configuration mode, confirm your configuration by entering the **show class-of-service** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
```

```
user@host# show class-of-service
interfaces {
irb {
  unit 0 {
  rewrite-rules {
  dscp rewrite-dscps;
    }
  }
}
}
rewrite-rules {
  dscp rewrite-dscps {
  forwarding-class be-class {
  loss-priority low code-point 000000;
    loss-priority high code-point 000001;
    }
    forwarding-class ef-class {
      loss-priority low code-point 101110;
    loss-priority high code-point 101111;
  }
  forwarding-class af-class {
    loss-priority low code-point 001010;
  loss-priority high code-point 001100;
}
forwarding-class nc-class {
  loss-priority low code-point 110000;
  loss-priority high code-point 110001;
  }
}
}
```

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

## Verification

### Verifying Rewrite Rules Configuration

**Purpose** Verify that rewrite rules are configured properly.

**Action** From configuration mode, enter the **show class-of-service** command.

```
user@host> show class-of-service
Physical interface: irb, Index: 130
 Maximum usable queues: 8, Queues in use: 4
 Scheduler map: <default> , Index: 2
Congestion-notification: Disabled

Logical interface: irb.10, Index: 71
Object                 Name                  Type                  Index
Classifier             ipprec-compatibility  ip                       13
```

**Meaning** Rewrite rules are configured on IRB interface as expected.

**Related Documentation**

- Rewrite Rules Overview on page 211

## Code-Point Aliases Overview

**Supported Platforms** SRX Series, vSRX

A code-point alias assigns a name to a pattern of code-point bits. You can use this name instead of the bit pattern when you configure other class-of-service (CoS) components, such as classifiers, drop-profile maps, and rewrite rules.

When you configure classes and define classifiers, you can refer to the markers by alias names. You can configure user-defined classifiers in terms of alias names. If the value of an alias changes, it alters the behavior of any classifier that references it.

The following types of code points are supported by Junos operating system (OS):

- DSCP—Defines aliases for DiffServ code point (DSCP) IPv4 values.

  You can refer to these aliases when you configure classes and define classifiers.
- DSCP-IPv6—Defines aliases for DSCP IPv6 values.

  You can refer to these aliases when you configure classes and define classifiers.
- EXP—Defines aliases for MPLS EXP bits.

  You can map MPLS EXP bits to the device forwarding classes.
- inet-precedence—Defines aliases for IPv4 precedence values.

  Precedence values are modified in the IPv4 type-of-service (ToS) field and mapped to values that correspond to levels of service.

**Related Documentation**

- Default CoS Values and Aliases on page 217
- Example: Defining Code-Point Aliases for Bits on a Security Device on page 220

## Default CoS Values and Aliases

**Supported Platforms** SRX Series, vSRX

Table 30 on page 218 shows the default mapping between the standard aliases and the bit values.

### Table 30: Standard CoS Aliases and Bit Values

| CoS Value Type | Alias | Bit Value |
|---|---|---|
| MPLS EXP | be | 000 |
| | be1 | 001 |
| | ef | 010 |
| | ef1 | 011 |
| | af11 | 100 |
| | af12 | 101 |
| | nc1/cs6 | 110 |
| | nc2/cs7 | 111 |

Table 30: Standard CoS Aliases and Bit Values *(continued)*

| CoS Value Type | Alias | Bit Value |
|---|---|---|
| DSCP and DSCP IPv6 | ef | 101110 |
| | af11 | 001010 |
| | af12 | 001100 |
| | af13 | 001110 |
| | af21 | 010010 |
| | af22 | 010100 |
| | af23 | 010110 |
| | af31 | 011010 |
| | af32 | 011100 |
| | af33 | 011110 |
| | af41 | 100010 |
| | af42 | 100100 |
| | af43 | 100110 |
| | be | 000000 |
| | cs1 | 001000 |
| | cs2 | 010000 |
| | cs3 | 011000 |
| | cs4 | 100000 |
| | cs5 | 101000 |
| | nc1/cs6 | 110000 |
| | nc2/cs7 | 111000 |

Table 30: Standard CoS Aliases and Bit Values *(continued)*

| CoS Value Type | Alias | Bit Value |
|---|---|---|
| IEEE 802.1 | be | 000 |
| | be1 | 001 |
| | ef | 010 |
| | ef1 | 011 |
| | af11 | 100 |
| | af12 | 101 |
| | nc1/cs6 | 110 |
| | nc2/cs7 | 111 |
| IP precedence | be | 000 |
| | be1 | 001 |
| | ef | 010 |
| | ef1 | 011 |
| | af11 | 100 |
| | af12 | 101 |
| | nc1/cs6 | 110 |
| | nc2/cs7 | 111 |

**Related Documentation**

- Code-Point Aliases Overview on page 217
- Example: Defining Code-Point Aliases for Bits on a Security Device on page 220

## Example: Defining Code-Point Aliases for Bits on a Security Device

**Supported Platforms** SRX Series, vSRX

This example shows how to define code-point aliases for bits on a device.

- Requirements on page 221
- Overview on page 221

### Requirements

Before you begin, determine which default mapping to use. See "Default CoS Values and Aliases" on page 217.

### Overview

In this example, you configure class of service and specify names and values for the CoS code-point aliases that you want to configure. Finally, you specify CoS value using the appropriate formats.

### Configuration

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To define code-point aliases for bits on a device:

1. Configure class of service.

   ```
   [edit]
   user@host# edit class-of-service
   ```

2. Specify CoS values.

   ```
   [edit class-of-service]
   user@host# set code-point-aliases dscp my1 110001
   user@host# set code-point-aliases dscp my2 101110
   user@host# set code-point-aliases dscp be 000001
   user@host# set code-point-aliases dscp cs7 110000
   ```

3. If you are done configuring the device, commit the configuration.

   ```
   [edit]
   user@host# commit
   ```

### Verification

To verify the configuration is working properly, enter the **show class-of-service code-point-aliases dscp** command.

**Related Documentation**

- Code-Point Aliases Overview on page 217

## Schedulers Overview

**Supported Platforms** SRX Series, vSRX

You use schedulers to define the properties of output queues. These properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, the priority of the queue, and the random early detection (RED) drop profiles associated with the queue.

You associate the schedulers with forwarding classes by means of scheduler maps. You can then associate each scheduler map with an interface, thereby configuring the hardware queues, packet schedulers, and RED processes that operate according to this mapping.

An individual device interface has multiple queues assigned to store packets temporarily before transmission. To determine the order to service the queues, the device uses a round-robin scheduling method based on priority and the queue's weighted round-robin (WRR) credits. Junos OS schedulers allow you to define the priority, bandwidth, delay buffer size, rate control status, and RED drop profiles to be applied to a particular queue for packet transmission.

You can configure per-unit scheduling (also called logical interface scheduling) to allow multiple output queues on a logical interface and to associate an output scheduler with each queue.

NOTE: For Juniper Network devices, when configuring the *protocol parameter* in the **drop-profile-map** statement, TCP and non-TCP values are not supported; only the value *any* is supported.

This topic contains the following sections:



## Transmit Rate

The transmission rate determines the traffic transmission bandwidth for each forwarding class you configure. The rate is specified in bits per second (bps). Each queue is allocated some portion of the bandwidth of the outgoing interface.

This bandwidth amount can be a fixed value, such as 1 megabit per second (Mbps), a percentage of the total available bandwidth, or the rest of the available bandwidth. You can limit the transmission bandwidth to the exact value you configure, or allow it to exceed the configured rate if additional bandwidth is available from other queues (SRX5400, SRX5600, and SRX5800 devices do not support an exact value transmit rate). This property helps ensure that each queue receives the amount of bandwidth appropriate to its level of service.

The minimum transmit rate supported on high-speed interfaces is one-ten thousandth of the speed of that interface. For example, on a Gigabit Ethernet interface with a speed of 1000 Mbps, the minimum transmit rate is 100 Kbps (1,000 Mbps x 1/10,000). You can

configure transmit rates in the range 3200 bps through 160,000,000,000 bps. When the configured rate is less than the minimum transmit rate, the minimum transmit rate is used instead.

NOTE: Interfaces with slower interface speeds, like T1, E1, or channelized T1/E1/ISDN PRI, cannot support minimum transmit rates because the minimum transmit rate supported on a device is 3,200 bps.

Transmit rate assigns the weighted round-robin (WRR) priority values within a given priority level and not between priorities.

The transmit rate defines the transmission rate of a scheduler. The transmit rate determines the traffic bandwidth from each forwarding class you configure.

By default, queues 0 through 7 have the following percentage of transmission capacity:

- Queue 0—95 percent
- Queue 1—0 percent
- Queue 2—0 percent
- Queue 3—0 percent
- Queue 4—0 percent
- Queue 6—0 percent
- Queue 7—5 percent

To define a transmit rate, select the appropriate option:

- To specify a transmit rate, select **rate** and type an integer from 3200 to 160,000,000,000 bits per second.
- To enforce an exact transmit rate, select **rate**.
- To specify the remaining transmission capacity, select **remainder**.
- To specify a percentage of transmission capacity, select **percent** and type an integer from 1 through 100.

Optionally, you can specify the percentage of the remainder to be used for allocating the transmit rate of the scheduler on a prorated basis. If there are still points left even after allocating the remainder percentage with the transmit rate and there are no queues, then the points are allocated point by point to each queue in a round-robin method. If the remainder percentage is not specified, the remainder value will be shared equally.

## Delay Buffer Size

You can configure the delay buffer size to control congestion at the output stage. A delay buffer provides packet buffer space to absorb burst traffic up to a specified duration of delay. When the buffer is full, all packets are dropped.

On Juniper Networks devices, you can configure larger delay buffers on channelized T1/E1 interfaces. Larger delay buffers help these slower interfaces to avoid congestion and packet dropping when they receive large bursts of traffic.

By default, SRX300, SRX320, SRX340, SRX345, and SRX550M device interfaces support a delay buffer time of 100,000 microseconds. (Platform support depends on the Junos OS release in your implementation.)

To define a delay buffer size for a scheduler, select the appropriate option:

- To enforce exact buffer size, select **Exact**.
- To specify a buffer size as a temporal value (microseconds), select **Temporal**.
- To specify buffer size as a percentage of the total buffer, select **Percent** and type an integer from 1 through 100.
- To specify buffer size as the remaining available buffer, select **Remainder**.

Optionally, you can specify the percentage of the remainder to be used for allocating the buffer size of the scheduler on a prorated basis.

By default, sizes of the delay buffer queues 0 through 7 have the following percentage of the total available buffer space:

- Queue 0—95 percent
- Queue 1—0 percent
- Queue 2—0 percent
- Queue 3—0 percent
- Queue 4—0 percent
- Queue 5—0 percent
- Queue 6—0 percent
- Queue 7—5 percent

NOTE: A large buffer size value correlates with a greater possibility of packet delays. This might not be practical for sensitive traffic such as voice or video.

NOTE: For a Juniper Networks device, if the buffer size percentage is set to zero for T1 interfaces, traffic does not pass.

## Scheduling Priority

Scheduling priority determines the order in which an output interface transmits traffic from the queues, thus ensuring that queues containing important traffic are provided better access to the outgoing interface.

The queues for an interface are divided into sets based on their priority. Each set contains queues of the same priority. The device examines the sets in descending order of priority. If at least one queue in a set has a packet to transmit, the device selects that set. If multiple queues in the set have packets to transmit, the device selects a queue from the set according to the weighted round-robin (WRR) algorithm that operates within the set.

The packets in a queue are transmitted based on the configured scheduling priority, the transmit rate, and the available bandwidth.

The scheduling priority of the scheduler determines the order in which an output interface transmits traffic from the queues. You can set scheduling priority at different levels in an order of increasing priority from low to high. A high-priority queue with a high transmission rate might lock out lower-priority traffic.

To specify a scheduling priority, select one of the following levels:

- **high**—Packets in this queue have high priority.
- **low**—Packets in this queue are transmitted last.
- **medium—low**—Packets in this queue have medium-low priority.
- **medium—high**—Packets in this queue have medium-high priority.
- **strict—high**—Packets in this queue are transmitted first.

## Shaping Rate

Shaping rates control the maximum rate of traffic transmitted on an interface. You can configure the shaping rate so that the interface transmits less traffic than it is physically capable of carrying.

You can configure shaping rates on logical interfaces. By default, output scheduling is not enabled on logical interfaces. Logical interface scheduling (also called per-unit scheduling) allows you to enable multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.

By default, the logical interface bandwidth is the average of unused bandwidth for the number of logical interfaces that require default bandwidth treatment. You can specify a peak bandwidth rate in bits per second (bps), either as a complete decimal number or as a decimal number followed by the abbreviation *k* (1000), *m* (1,000,000), or *g* (1,000,000,000). The range is from 1000 through 32,000,000,000 bps.

For low-speed interfaces, the queue-limit values might become lower than the interface MTU so that traffic with large packets can no longer pass through some of the queues. If you want larger-sized packets to flow through, set the buffer-size configuration in the scheduler to a larger value. For more accuracy, the 100-ms queue-limit values are calculated based on shaping rate and not on interface rates.

The shaping rate defines the minimum bandwidth allocated to a queue. The default shaping rate is 100 percent, which is the same as no shaping at all. To define a shaping rate, select the appropriate option:

- To specify shaping rate as an absolute number of bits per second, select **rate** and type an integer from 3200 to 160,000,000,000 bits per second.
- To specify shaping rate as a percentage, select **percent** and type an integer from 0 through 100.

**Related Documentation**

- *Default Scheduler Settings*
- Example: Configuring Class-of-Service Schedulers on a Security Device on page 226
- *Scheduler Buffer Size Overview*
- *Example: Configuring a Large Delay Buffer on a Channelized T1 Interface*
- *Example: Configuring and Applying Scheduler Maps*
- *Transmission Scheduling Overview*

## Example: Configuring Class-of-Service Schedulers on a Security Device

**Supported Platforms** SRX Series, vSRX

This example shows how to configure CoS schedulers on a device.



### Requirements

Before you begin, determine the buffer size allocation method to use. See *Scheduler Buffer Size Overview*.

### Overview

An individual device interface has multiple queues assigned to store packets temporarily before transmission. To determine the order in which to service the queues, the device uses a round-robin scheduling method based on priority and the queue's weighted round-robin (WRR) credits. Junos OS schedulers allow you to define the priority, bandwidth, delay buffer size, rate control status, and RED drop profiles to be applied to a particular queue for packet transmission.

You configure schedulers to assign resources, priorities, and drop profiles to output queues. By default, only queues 0 and 3 have resources assigned.

NOTE: **Juniper Network devices support hierarchical schedulers, including per-unit schedulers.**

In this example, you configure a best-effort scheduler called be-scheduler. You set the priority as low and the buffer size to 40. You set the be-scheduler transmit-rate remainder percentage to 40. You configure an expedited forwarding scheduler called ef-scheduler and set the priority as high and the buffer size to 10. You set the ef-scheduler transmit-rate remainder percentage to 50.

Then you configure an assured forwarding scheduler called af-scheduler and set the priority as high and buffer size to 45. You set an assured forwarding scheduler transmit rate to 45. You then configure a drop profile map for assured forwarding as low and high priority. (DiffServ can have a RED drop profile associated with assured forwarding.)

Finally, you configure a network control scheduler called nc-scheduler and set the priority as low and buffer size to 5. You set a network control scheduler transmit rate to 5.

Table 31 on page 227 shows the schedulers created in this example.

### Table 31: Sample Schedulers

| Scheduler | For CoS Traffic Type | Assigned Priority | Allocated Portion of Queue Buffer | Allocated Portion of Remainder (Transmit Rate) |
|---|---|---|---|---|
| **be-scheduler** | Best-effort traffic | Low | 40 percent | 40 percent |
| **ef-scheduler** | Expedited forwarding traffic | High | 10 percent | 50 percent |
| **af-scheduler** | Assured forwarding traffic | High | 45 percent | — |
| **nc-scheduler** | Network control traffic | Low | 5 percent | — |

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service schedulers be-scheduler priority low buffer-size percent 40
set class-of-service schedulers be-scheduler transmit-rate remainder 40
set class-of-service schedulers ef-scheduler priority high buffer-size percent 10
set class-of-service schedulers ef-scheduler transmit-rate remainder 50
set class-of-service schedulers af-scheduler priority high buffer-size percent 45
set class-of-service schedulers af-scheduler transmit-rate percent 45
set class-of-service schedulers af-scheduler drop-profile-map loss-priority low protocol
    any drop-profile af-normal
set class-of-service schedulers af-scheduler drop-profile-map loss-priority high protocol
    any drop-profile af-with-PLP
set class-of-service schedulers nc-scheduler priority low buffer-size percent 5
set class-of-service schedulers nc-scheduler transmit-rate percent 5
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure CoS schedulers:

1. Configure a best-effort scheduler.

```
[edit]
user@host# edit class-of-service schedulers be-scheduler
```

2. Specify a best-effort scheduler priority and buffer size.

```
[edit class-of-service schedulers be-scheduler]
user@host# set priority low
user@host# set buffer-size percent 40
```

3. Configure a remainder option for a best-effort scheduler transmit rate.

```
[edit class-of-service schedulers be-scheduler]
user@host# set transmit-rate remainder 40
```

4. Configure an expedited forwarding scheduler.

```
[edit]
user@host# edit class-of-service schedulers ef-scheduler
```

5. Specify an expedited forwarding scheduler priority and buffer size.

```
[edit class-of-service schedulers ef-scheduler]
user@host# set priority high
user@host# set buffer-size percent 10
```

6. Configure a remainder option for an expedited forwarding scheduler transmit rate.

```
[edit class-of-service schedulers ef-scheduler]
user@host# set transmit-rate remainder 50
```

7. Configure an assured forwarding scheduler.

```
[edit]
user@host# edit class-of-service schedulers af-scheduler
```

8. Specify an assured forwarding scheduler priority and buffer size.

```
[edit class-of-service schedulers af-scheduler]
user@host# set priority high
user@host# set buffer-size percent 45
```

9. Configure an assured forwarding scheduler transmit rate.

```
[edit class-of-service schedulers af-scheduler]
user@host# set transmit-rate percent 45
```

10. Configure a drop profile map for assured forwarding low and high priority.

```
[edit class-of-service schedulers af-scheduler]
user@host# set drop-profile-map loss-priority low protocol any drop-profile
  af-normal
user@host# set drop-profile-map loss-priority high protocol any drop-profile
  af-with-PLP
```

11. Configure a network control scheduler.

```
[edit]
user@host# edit class-of-service schedulers nc-scheduler
```

12. Specify a network control scheduler priority and buffer size.

```
[edit class-of-service schedulers nc-scheduler]
user@host# set priority low
user@host# set buffer-size percent 5
```

13. Configure a network control scheduler transmit rate.

```
[edit class-of-service schedulers nc-scheduler]
user@host# set transmit-rate percent 5
```

**Results** From configuration mode, confirm your configuration by entering the **show class-of-service** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host# show class-of-service
  schedulers {
  be-scheduler {
  transmit-rate remainder 40;
    buffer-size percent 40;
  priority low;
  }
  ef-scheduler {
    transmit-rate remainder 50;
    buffer-size percent 10;
  priority high;
  }
  af-scheduler {
    transmit-rate percent 45;
    buffer-size percent 45;
    priority high;
      drop-profile-map loss-priority low protocol any drop-profile af-normal;
    drop-profile-map loss-priority high protocol any drop-profile af-with-PLP;
    }
    nc-scheduler {
      transmit-rate percent 5;
      buffer-size percent 5;
    priority low;
  }
```

```
}
```

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

## Verification

### Verifying Schedulers Configuration

**Purpose** Verify that the schedulers are configured properly.

**Action** From operational mode, enter the **show class-of-service** command.

**Related Documentation**

- Schedulers Overview on page 221
- *Default Scheduler Settings*
- *Example: Configuring a Large Delay Buffer on a Channelized T1 Interface*
- *Example: Configuring and Applying Scheduler Maps*
- *Transmission Scheduling Overview*

# Example: Configuring a Large Delay Buffer on a Security Device IRB Interface

**Supported Platforms** SRX1500

This example shows how to configure a large delay buffer on an IRB interface to help slower interfaces avoid congestion and packet dropping when they receive large bursts of traffic.



## Requirements

Before you begin, enable the large buffer feature on the IRB interface and then configure a buffer size for each queue in the CoS scheduler. See *Scheduler Buffer Size Overview*.

## Overview

On devices, you can configure large delay buffers on an irb interfaces.

In this example, you configure scheduler map to associate schedulers to a defined forwarding class **be-class**, **ef-class** , **af-class**, and **nc-class** using scheduler map **large-buf-sched-map**. You apply scheduler maps to irb interface, and define per-unit scheduler for the IRB interface.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service scheduler-maps large-buf-sched-map forwarding-class be-class
  scheduler be-scheduler
set class-of-service scheduler-maps large-buf-sched-map forwarding-class ef-class
  scheduler ef-scheduler
set class-of-service scheduler-maps large-buf-sched-map forwarding-class af-class
  scheduler af-scheduler
set class-of-service scheduler-maps large-buf-sched-map forwarding-class nc-class
  scheduler nc-scheduler
set class-of-service interfaces irb unit 0 scheduler-map large-buf-sched-map
set interfaces irb per-unit-scheduler
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a large delay buffer on a channelized T1 interface:

1. Configure the scheduler map to associate schedulers with defined forwarding classes.

```
[edit class-of-service]
set scheduler-maps large-buf-sched-map forwarding-class be-class scheduler
  be-scheduler
set scheduler-maps large-buf-sched-map forwarding-class ef-class scheduler
  ef-scheduler
set scheduler-maps large-buf-sched-map forwarding-class af-class scheduler
  af-scheduler
set scheduler-maps large-buf-sched-map forwarding-class nc-class scheduler
  nc-scheduler
```

2. Apply the scheduler map to the IRB interface.

```
[edit ]
user@host# set interfaces irb unit 0 scheduler-map large-buf-sched-map
```

3. Define the per-unit scheduler for the irb interface.

```
[edit ]
user@host# set interfaces irb per-unit-scheduler
```

**Results**

From configuration mode, confirm your configuration by entering the **show class-of-service** and **show chassis** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
  user@host# show class-of-service
  interfaces {
  irb {
    unit 0 {
      scheduler-map large-buf-sched-map;
    }
  }
}
scheduler-maps {
  large-buf-sched-map {
    forwarding-class be-class scheduler be-scheduler;
    forwarding-class ef-class scheduler ef-scheduler;
    forwarding-class af-class scheduler af-scheduler;
    forwarding-class nc-class scheduler nc-scheduler;
  }
}
```

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

## Verification

### Verifying Large Delay Buffers Configuration

**Purpose** Verify that the large delay buffers are configured properly.

**Action** From configuration mode, enter the **show class-of-service interface irb** command.

```
user@host> show class-of-service interface irb


Physical interface: irb, Index: 132
Maximum usable queues: 8, Queues in use: 4Code point type: dscp
Scheduler map: <default>, Index :2
Congestion-notification: Disabled
Logical interface: irb.10, Index: 73
Object                  Name                    Type                    Index
Classifier              ipprec-compatibility    ip                         13
```

**Meaning** The large delay buffers are configured on IRB interface as expected.

**Related Documentation**

- Schedulers Overview on page 221
- *Default Scheduler Settings*
- *Example: Configuring Class-of-Service Schedulers*
- *Example: Configuring and Applying Scheduler Maps*
- *Transmission Scheduling Overview*

## Virtual Channels Overview

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, vSRX

You can configure virtual channels to limit traffic sent from a corporate headquarters to its branch offices. Virtual channels might be required when the headquarters site has an expected aggregate bandwidth higher than that of the individual branch offices. The headquarters router must limit the traffic sent to each branch office router to avoid oversubscribing their links. For instance, if branch 1 has a 1.5 Mbps link and the headquarters router attempts to send 6 Mbps to branch 1, all of the traffic in excess of 1.5-Mbps is dropped in the ISP network.

You configure virtual channels on a logical interface. Each virtual channel has a set of eight queues with a scheduler and an optional shaper. You can use an output firewall filter to direct traffic to a particular virtual channel. For example, a filter can direct all traffic with a destination address for branch office 1 to virtual channel 1, and all traffic with a destination address for branch office 2 to virtual channel 2.

Although a virtual channel group is assigned to a logical interface, a virtual channel is quite different from a logical interface. The only features supported on a virtual channel are queuing, packet scheduling, and accounting. Rewrite rules and routing protocols apply to the entire logical interface.

When you configure virtual channels on an interface, the virtual channel group uses the same scheduler and shaper you configure at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level. In this way, virtual channels are an extension of regular scheduling and shaping and are not independent entities.

**Related Documentation**

- Understanding Virtual Channels on page 233
- Example: Configuring Virtual Channels on a Security Device on page 235

## Understanding Virtual Channels

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, vSRX

You configure a virtual channel to set up queuing, packet scheduling, and accounting rules to be applied to one or more logical interfaces. You must apply then the virtual channel to a particular logical interface.

You also create a list of virtual channels that you can assign to a virtual channel group. To define a virtual channel group that you can assign to a logical interface, include the **virtual-channel-groups** statement at the **[edit class-of-service]** hierarchy level.

The *virtual-channel-group-name* can be any name that you want. The *virtual-channel-name* must be one of the names that you define at the **[edit class-of-service virtual-channels]** hierarchy level. You can include multiple virtual channel names in a group.

The scheduler map is required. The *map-name* must be one of the scheduler maps that you configure at the **[edit class-of-service scheduler-maps]** hierarchy level. For more

information, see "Example: Configuring Class-of-Service Schedulers on a Security Device" on page 226.

The shaping rate is optional. If you configure the shaping rate as a percentage, when the virtual channel is applied to a logical interface, the shaping rate is set to the specified percentage of the interface bandwidth. If you configure a shaper on a virtual channel, the shaper limits the maximum bandwidth transmitted by that virtual channel. Virtual channels without a shaper can use the full logical interface bandwidth. If there are multiple unshaped virtual channels, they share the available logical interface bandwidth equally.

When you apply the virtual channel group to a logical interface, a set of eight queues is created for each of the virtual channels in the group. The **scheduler-map** statement applies a scheduler to these queues. If you include the **shaping-rate** statement, a shaper is applied to the entire virtual channel.

You must configure one of the virtual channels in the group to be the default channel. Therefore, the **default** statement is required in the configuration of one virtual channel per channel group. Any traffic not explicitly directed to a particular channel is transmitted by this default virtual channel.

For the corresponding physical interface, you must also include the **per-unit-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level as follows:

```
[edit interfaces interface-name]
per-unit-scheduler;
```

The **per-unit-scheduler** statement enables one set of output queues for each logical interface configured under the physical interface.

When you apply a virtual channel group to a logical interface, the software creates a set of eight queues for each of the virtual channels in the group.

If you apply a virtual channel group to multiple logical interfaces, the software creates a set of eight queues on each logical interface. The virtual channel names listed in the group are used on all the logical interfaces. We recommend specifying the scheduler and shaping rates in the virtual channel configuration in terms of percentages, rather than absolute rates. This allows you to apply the same virtual channel group to logical interfaces that have different bandwidths.

When you apply a virtual channel group to a logical interface, you cannot include the **scheduler-map** and **shaping-rate** statements at the [edit class-of-service interfaces *interface-name* unit *logical-unit-number*] hierarchy level. In other words, you can configure a scheduler map and a shaping rate on a logical interface, or you can configure virtual channels on the logical interface, but not both.

If you configure multiple logical interfaces on a single physical interface, each logical interface is guaranteed an equal fraction of the physical interface bandwidth as follows:

```
logical-interface-bandwidth =
physical-interface-bandwidth / number-of-logical-interfaces
```

If one or more logical interfaces do not completely use their allocation, the other logical interfaces share the excess bandwidth equally.

If you configure multiple virtual channels on a logical interface, they are each guaranteed an equal fraction of the logical interface bandwidth as follows:

```
virtual-channel-bandwidth =
logical-interface-bandwidth / number-of-virtual-channels
```

If you configure a shaper on a virtual channel, the shaper limits the maximum bandwidth transmitted by that virtual channel. Virtual channels without a shaper can use the full logical interface bandwidth. If there are multiple unshaped virtual channels, they share the available logical interface bandwidth equally.

**Related Documentation**

- Virtual Channels Overview on page 233
- Example: Configuring Virtual Channels on a Security Device on page 235

## Example: Configuring Virtual Channels on a Security Device

**Supported Platforms** SRX1500

This example shows how to create virtual channels between a headquarters and its branch office.



### Requirements

Before you begin, ensure that your headquarters and branch office have a network connection where the expected aggregate bandwidth is higher for your headquarters than for your branch office. The devices at your headquarters will then be set up to limit the traffic sent to the branch office to avoid oversubscribing the link.

### Overview

In this example, you create the virtual channels as branch1–vc, branch2–vc, branch3–vc, and default-vc. You then define the virtual channel group as wan-vc-group to include the four virtual channels and assign the scheduler map as bestscheduler to each virtual channel. Three of the virtual channels are shaped to 1.5 Mbps. The fourth virtual channel is default-vc, and it is not shaped. Hence can use the full interface bandwidth.

Then you apply them in the firewall filter as choose-vc to the device's irb interface. The output filter on the interface sends all traffic with a destination address matching 192.168.10.0/24 to branch1-vc, and similar configurations are set for branch2-vc and branch3-vc. Traffic not matching any of the addresses goes to the default, unshaped virtual channel.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service virtual-channels branch1-vc
set class-of-service virtual-channels branch2-vc
set class-of-service virtual-channels branch3-vc
set class-of-service virtual-channels default-vc
set class-of-service virtual-channel-groups wan-vc-group branch1-vc scheduler-map
  bestscheduler
set class-of-service virtual-channel-groups wan-vc-group branch2-vc scheduler-map
  bestscheduler
set class-of-service virtual-channel-groups wan-vc-group branch3-vc scheduler-map
  bestscheduler
set class-of-service virtual-channel-groups wan-vc-group default-vc scheduler-map
  bestscheduler
set class-of-service virtual-channel-groups wan-vc-group default-vc default
set class-of-service virtual-channel-groups wan-vc-group branch1-vc shaping-rate
  1500000
set class-of-service virtual-channel-groups wan-vc-group branch2-vc shaping-rate
  1500000
set class-of-service virtual-channel-groups wan-vc-group branch3-vc shaping-rate
  1500000
set class-of-service interfaces t3-1/0/0 unit 0 virtual-channel-group wan-vc-group
set firewall family inet filter choose-vc term branch1 from destination-address
  192.168.10.0/24
set firewall family inet filter choose-vc term branch1 then accept
set firewall family inet filter choose-vc term branch1 then virtual-channel branch1-vc
set firewall family inet filter choose-vc term branch1 then virtual-channel branch2-vc
set firewall family inet filter choose-vc term branch1 then virtual-channel branch3-vc
set interfaces irb unit 0 family inet filter output choose-vc
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure virtual channels:

1. Define the virtual channels and the default virtual channel.

   ```
   [edit]
   user@host# edit class-of-service
   user@host# set virtual-channels branch1-vc
   user@host# set virtual-channels branch2-vc
   user@host# set virtual-channels branch3-vc
   user@host# set virtual-channels default-vc
   ```

2. Define the virtual channel group and assign each virtual channel a scheduler map.

   ```
   [edit class-of-service]
   ```

```
user@host# set virtual-channel-groups wan-vc-group branch1-vc scheduler-map
  bestscheduler
user@host# set virtual-channel-groups wan-vc-group branch2-vc scheduler-map
  bestscheduler
user@host# set virtual-channel-groups wan-vc-group branch3-vc scheduler-map
  bestscheduler
user@host# set virtual-channel-groups wan-vc-group default-vc scheduler-map
  bestscheduler
user@host# set virtual-channel-groups wan-vc-group default-vc default
```

3. Specify a shaping rate.

```
[edit class-of-service]
user@host# set virtual-channel-groups wan-vc-group branch1-vc shaping-rate 1.5m
user@host# set virtual-channel-groups wan-vc-group branch2-vc shaping-rate
  1.5m
user@host# set virtual-channel-groups wan-vc-group branch3-vc shaping-rate
  1.5m
```

4. Apply the virtual channel group to the irb interface.

```
[edit class-of-service]
user@host# set interfaces irb unit 0 virtual-channel-group wan-vc-group
```

5. Create the firewall filter to select the traffic.

```
[edit firewall]
user@host# set family inet filter choose-vc term branch1 from destination
  192.168.10.0/24
user@host# set family inet filter choose-vc term branch1 then accept
user@host# set family inet filter choose-vc term branch1 then virtual-channel
  branch1-vc
user@host# set family inet filter choose-vc term branch1 then virtual-channel
  branch2-vc
user@host# set family inet filter choose-vc term branch1 then virtual-channel
  branch3-vc
```

6. Apply the firewall filter to output traffic.

```
[edit interfaces]
user@host# set irb unit 0 family inet filter output choose-vc
```

**Results** From configuration mode, confirm your configuration by entering the **show class-of-service**, **show firewall**, and **show interfaces irb** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show class-of-service
virtual-channels {
  branch1-vc;
  branch2-vc;
  branch3-vc;
  default-vc;
```

```
}
virtual-channel-groups {
    wan-vc-group {
        branch1-vc {
        scheduler-map bestscheduler;
        shaping-rate 1500000;
    }
    branch2-vc {
        scheduler-map bestscheduler;
        shaping-rate 1500000;
    }
    branch3-vc {
        scheduler-map bestscheduler;
        shaping-rate 1500000;
    }
    default-vc {
        scheduler-map bestscheduler;
        default;
    }
}
}
interfaces {
    irb {
        unit 0 {
            virtual-channel-group wan-vc-group;
        }
    }
}
[edit]
user@host# show firewall
family inet {
    filter choose-vc {
        term branch1 {
            from {
                destination-address {
                    192.168.10.0/24;
                }
            }
            then {
                virtual-channel branch3-vc;
                accept;
            }
        }
    }
}
[edit]
user@host# show interfaces irb
unit 0 {
    family inet {
        filter {
            output choose-vc;
        }
    }
}
```

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

## Verification

### Verifying Virtual Channel Configuration

**Purpose** Verify that the virtual channels are properly configured.

**Action** From configuration mode, enter the **show class-of-service**, **show firewall**, and **show interfaces irb** commands.

**Related Documentation**

- Virtual Channels Overview on page 233
- Understanding Virtual Channels on page 233

CHAPTER 19

# Configuring Layer 2 Switching Mode Chassis Clusters



## Layer 2 Ethernet Switching Capability in a Chassis Cluster Mode

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

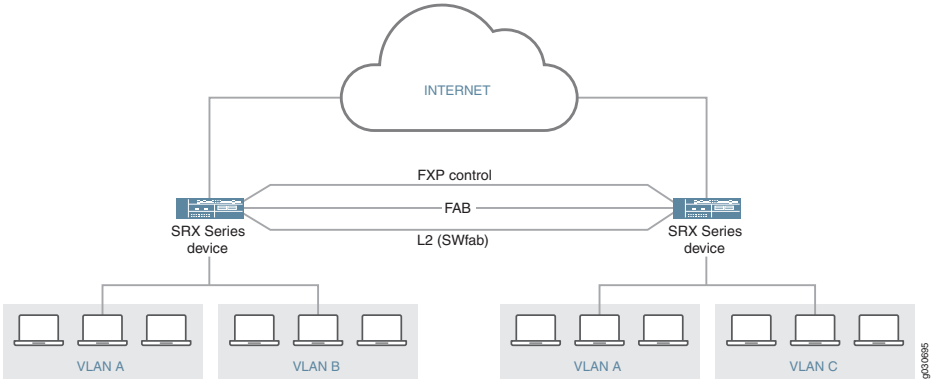


### Understanding Layer 2 Ethernet Switching Capability in a Chassis Cluster on SRX Series Devices

Ethernet ports support various Layer 2 features such as spanning-tree protocols (STPs), IEEE 802.1x, Link Layer Discovery Protocol (LLDP), and Multiple VLAN Registration Protocol (MVRP). With the extension of Layer 2 switching capability to devices in a chassis cluster, you can use Ethernet switching features on both nodes of a chassis cluster. You can configure the Ethernet ports on either node for family Ethernet switching. You can also configure a Layer 2 VLAN domain with member ports from both nodes and the Layer 2 switching protocols on both devices.

Figure 10 on page 242 shows the Layer 2 switching across chassis cluster nodes.

Figure 10: Layer 2 Ethernet Switching Across Chassis Cluster Nodes




To ensure that Layer 2 switching works seamlessly across chassis cluster nodes, a dedicated physical link connecting the nodes is required. This type of link is called a *switching fabric interface*. Its purpose is to carry Layer 2 traffic between nodes.

NOTE: Configuring a LAG with members across nodes is not supported.

CAUTION: If a switching fabric interface is not configured on both nodes, and if you try to configure Ethernet switching-related features on the nodes, then the behavior of the nodes might be unpredictable.

## Understanding Chassis Cluster Failover and New Primary Election

When chassis cluster failover occurs, a new primary node is elected and the Ethernet switching process (eswd) runs in a different node. During failover, the chassis control subsystem is restarted. Also during failover, traffic outage occurs until the PICs are up and the VLAN entries are reprogrammed. After failover, all Layer 2 protocols reconverge because Layer 2 protocol states are not maintained in the secondary node.

NOTE: The Q-in-Q feature in chassis cluster mode is not supported because of chip limitation for swfab interface configuration in Broadcom chipsets.

**Related Documentation**

- Ethernet Switching and Layer 2 Transparent Mode Overview

# Example: Configuring Switch Fabric Interfaces to Enable Switching in Chassis Cluster Mode on a Security Device

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

This example shows how to configure switching fabric interfaces to enable switching in chassis cluster mode.



## Requirements

- The physical link used as the switch fabric member must be directly connected to the device. (this sentence is confusing)
- Switching fabric interfaces must be configured on ports that support switching features. See "Ethernet Ports Switching Overview for Security Devices" on page 104 for innformation about the ports on which switching features are supported.

The physical link used as the switch fabric member must be directly connected to the device. Switching supported ports must be used for switching fabric interfaces. See "Ethernet Ports Switching Overview for Security Devices" on page 104 for switching supported ports.

Before you begin, See *Example: Configuring the Chassis Cluster Fabric Interfaces*.

## Overview

In this example, pseudointerfaces swfab0 and swfab1 are created for Layer 2 fabric functionality. You also configure dedicated Ethernet ports on each node to be associated with the swfab interfaces.

## Configuration

**CLI Quick Configuration**

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

```
set interfaces swfab0 fabric-options member-interfaces ge-0/0/9
set interfaces swfab0 fabric-options member-interfaces ge-0/0/10
set interfaces swfab1 fabric-options member-interfaces ge-7/0/9
set interfaces swfab1 fabric-options member-interfaces ge-7/0/10
```

**Step-by-Step Procedure**

To configure swfab interfaces:

1. Configure swfab0 and swfab1 and associate these switch fabric interfaces to enable switching across the nodes. Note that swfab0 corresponds to node 0 and swfab1 corresponds to node 1.

```
{primary:node0} [edit]
user@host# set interfaces swfab0 fabric-options member-interfaces ge-0/0/9
user@host# set interfaces swfab0 fabric-options member-interfaces ge-0/0/10
user@host# set interfaces swfab1 fabric-options member-interfaces ge-7/0/9
user@host# set interfaces swfab1 fabric-options member-interfaces ge-7/0/10
```

2. If you are done configuring the device, commit the configuration.

```
{primary:node0} [edit]
user@host# commit
```

**Results** From configuration mode, confirm your configuration by entering the **show interfaces swfab0** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct the configuration.

```
[edit]
user@host# show interfaces swfab0
fabric-options{
   member-interfaces {
   ge-0/0/9;
   ge-0/0/10;
   }
}
```

## Verification

To confirm that the configuration is working properly, perform these tasks:

- Verifying Switching Fabric Ports on page 244

***Verifying Switching Fabric Ports***

**Purpose** Verify that you are able to configure multiple ports as members of switching fabric ports.

**Action** From configuration mode, enter the **show interfaces swfab0** command to view the configured interfaces for each port.

```
user@host# show interfaces swfab0
fabric-options{
 member-interfaces {
   ge-0/0/9;
   ge-0/0/10;
}
}
```

From configuration mode, enter the **show chassis cluster ethernet-switching interfaces** command to view the appropriate member interfaces.

```
user@host# run show chassis cluster ethernet-switching interfaces
swfab0:
    Name                Status
    ge-0/0/9              up
    ge-0/0/10             up
swfab1:
    Name                Status
    ge-7/0/9              up
    ge-7/0/10             up
```

**Related Documentation**
- *SRX Series Chassis Cluster Configuration Overview*

## Example: Configuring IRB and VLAN with Members Across Two Nodes on a Security Device

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M



### Requirements

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

### Overview

This example shows the configuration of integrated routing and bridging (IRB) and configuration of a VLAN with members across node 0 and node 1.

### Configuration

**CLI Quick Configuration**

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

```
set interfaces ge-0/0/3 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/3 unit 0 family ethernet-switching vlan members vlan100
set interfaces ge-0/0/4 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/4 unit 0 family ethernet-switching vlan members vlan100
set interfaces ge-7/0/5 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-7/0/5 unit 0 family ethernet-switching vlan members vlan100
set interfaces irb unit 100 family inet address 192.0.2.100/24
set vlans vlan100 vlan-id 100
set vlans vlan100 l3-interface irb.100
```

**Step-by-Step Procedure**

To configure IRB and a VLAN:

1. Configure Ethernet switching on the node0 interface.

```
{primary:node0} [edit]
user@host# set interfaces ge-0/0/3 unit 0 family ethernet-switching interface-mode
    access
user@host# set interfaces ge-0/0/4 unit 0 family ethernet-switching interface-mode
    access
```

2. Configure Ethernet switching on the node1 interface.

```
{primary:node0} [edit]
user@host# set interfaces ge-7/0/5 unit 0 family ethernet-switching interface-mode
  trunk
```

3. Create VLAN vlan100 with vlan-id 100.

```
{primary:node0} [edit]
user@host# set vlans vlan100 vlan-id 100
```

4. Add interfaces from both nodes to the VLAN.

```
{primary:node0} [edit]
user@host# set interfaces ge-0/0/3 unit 0 family ethernet-switching vlan members
  vlan100
user@host# set interfaces ge-0/0/4 unit 0 family ethernet-switching vlan members
  vlan100
user@host# set interfaces ge-7/0/5 unit 0 family ethernet-switching vlan members
  vlan100
```

5. Create an IRB logical interface.

```
user@host# set interfaces irb unit 100 family inet address 192.0.2.100/24
```

6. Associate an IRB interface with the VLAN.

```
user@host# set vlans vlan100 l3-interface irb.100
```

7. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

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

```
[edit]
user@host# show vlans
vlan100 {
  vlan-id 100;
  l3-interface irb.100;
}
[edit]
user@host# show interfaces
ge-0/0/3 {
  unit 0 {
    family ethernet-switching {
      interface-mode access;
      vlan {
        members vlan100;
      }
    }
  }
```

```
}
ge-0/0/4 {
  unit 0 {
    family ethernet-switching {
      interface-mode access;
      vlan {
        members vlan100;
      }
    }
  }
}
ge-7/0/5 {
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members vlan100;
      }
    }
  }
}
irb {
  unit 100 {
    family inet {
      address 192.0.2.100/24;
    }
  }
}
```

## Verification

### Verifying VLAN and IRB

**Purpose** Verify that the configurations of VLAN and IRB are working properly.

**Action** From operational mode, enter the **show interfaces terse ge-0/0/3** command to view the node 0 interface.

```
user@host> show interfaces terse ge-0/0/3
Interface               Admin Link Proto    Local                 Remote
ge-0/0/3                up    up
ge-0/0/3.0              up    up   eth-switch
```

From operational mode, enter the **show interfaces terse ge-0/0/4** command to view the node 0 interface.

```
user@host> show interfaces terse ge-0/0/4
Interface               Admin Link Proto    Local                 Remote
ge-0/0/4                up    up
ge-0/0/4.0              up    up   eth-switch
```

From operational mode, enter the **show interfaces terse ge-7/0/5** command to view the node1 interface.

```
user@host> show interfaces terse ge-7/0/5
Interface               Admin Link Proto    Local                 Remote
ge-7/0/5                up    up
ge-7/0/5.0              up    up   eth-switch
```

From operational mode, enter the **show vlans** command to view the VLAN interface.

```
user@host> show vlans
Routing instance      VLAN name     Tag      Interfaces
default-switch        default        1
default-switch        vlan100        100     ge-0/0/3.0*
                                             ge-0/0/4.0*
                                             ge-7/0/5.0*
```

From operational mode, enter the **show ethernet-switching interface** command to view the information about Ethernet switching interfaces.

```
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                         LH - MAC limit hit, DN - interface down,
                        MMAS - Mac-move action shutdown,  AS - Autostate-exclude
 enabled,
                         SCTL - shutdown by Storm-control )

Logical         Vlan                  TAG   MAC    STP         Logical
 Tagging
interface       members                     limit  state       interface flags
ge-0/0/3.0                                  16383              DN
 untagged
                vlan100               100   1024   Discarding
 untagged
ge-0/0/4.0                                  16383              DN
 untagged
                vlan100               100   1024   Discarding
 untagged
ge-7/0/5.0                                  16383              DN
 tagged
                vlan100               100   1024   Discarding
 tagged
```

**Meaning** The output shows the VLAN and IRB are configured and working fine.

**Related Documentation**

- Example: Configuring an IRB Interface

CHAPTER 20

# Configuring 802.1X Port-Based Network Authentication



## Understanding 802.1X Port-Based Network Authentication

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

NOTE: From Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D75 and Junos OS Release 17.3R1, IEEE 802.1X port-based network authentication is not supported.

NOTE: Starting in Junos OS 15.1X49-D80, 802.1X port-based authentication is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

Both IEEE 802.1X authentication and MAC RADIUS authentication both provide network edge security, protecting Ethernet LANs from unauthorized user access by blocking all traffic to and from devices at the interface until the supplicant's credential or MAC address is presented and matched on the *authentication server* (a RADIUS server). When the supplicant is authenticated, the switch stops blocking access and opens the interface to the supplicant.

A LAN network configured for 802.1X authentication contains three basic components:

- **Supplicant**—The IEEE term for a host that requests to join the network. The host can be responsive or nonresponsive. A responsive host is one on which 802.1X authentication is enabled and that provides authentication credentials (such as a user name and password). A nonresponsive host is one on which 802.1X authentication is not enabled.

- **Authenticator port access entity**—The IEEE term for the authenticator. The SRX Series device is the authenticator and controls access by blocking all traffic from host/supplicant until they are authenticated.
- **Authentication server**—The server containing the back-end database that makes authentication decisions. (Junos OS supports RADIUS authentication servers.) The authentication server contains credential information for each supplicant that can connect to the network. The authenticator forwards credentials supplied by the supplicant to the authentication server. If the credentials forwarded by the authenticator match the credentials in the authentication server database, access is granted. If the credentials forwarded do not match, access is denied.

Table 32 on page 250 lists the features that the implementation of 802.1X authentication provides for specific devices. (Platform support depends on the Junos OS release in your installation.). Table 33 on page 250 lists the supplicant capacities that the implementation of 802.1X authentication provides for specific devices.

### Table 32: 802.1X Authentication Features

| Feature | SRX300/SRX320 | SRX340/SRX345 | SRX550M | SRX1500 |
|---|---|---|---|---|
| Dynamic VLAN assignment | Yes | Yes | Yes | Yes |
| MAC RADIUS authentication | Yes | Yes | Yes | Yes |
| Static MAC bypass | Yes | Yes | Yes | Yes |
| Guest VLAN | Yes | Yes | Yes | Yes |
| RADIUS server failure fallback | Yes | Yes | Yes | Yes |
| VoIP VLAN support | Yes | Yes | Yes | Yes |
| RADIUS accounting | Yes | Yes | Yes | Yes |

### Table 33: 802.1x Supplicant Capacities

| Capacities | SRX300/SRX320 | SRX340/SRX345 | SRX550M | SRX1500 |
|---|---|---|---|---|
| Supplicants per port | 64 | 64 | 64 | 64 |
| Supplicants per system | 2K | 2K | 2K | 2K |
| Supplicants with dynamic VLAN assignments | 64 | 300 | 2K | 2K |

This topic contains the following sections:



## Dynamic VLAN Assignment

When a supplicant first connects to an SRX Series device, the authenticator sends a request to the supplicant to begin 802.1X authentication. If the supplicant is an 802.1X-enabled device, it responds, and the authenticator relays an authentication request to the RADIUS server.

As part of the reply to the authentication request, the RADIUS server returns information about the VLAN to which the port belongs. By configuring the VLAN information at the RADIUS server, you can control the VLAN assignment on the port.

## MAC RADIUS Authentication

If the authenticator sends three requests to a supplicant to begin 802.1X authentication and receives no response, the supplicant is considered nonresponsive. For a nonresponsive supplicant, the authenticator sends a request to the RADIUS server for authentication of the supplicant's MAC address. If the MAC address matches an entry in a predefined list of MAC addresses on the RADIUS server, authentication is granted and the authenticator opens LAN access on the interface where the supplicant is connected.

You can configure the number of times the authenticator attempts to receive a response and the time period between attempts.

## Static MAC Bypass

The authenticator can allow particular supplicants direct access to the LAN, bypassing the authentication server, by including the supplicants' MAC addresses in the static MAC bypass list configured on the SRX Series device. Supplicants' MAC addresses are first checked against this list. If a match is found, the corresponding supplicant is considered successfully authenticated and the interface is opened up for it. No further authentication is done for that supplicant. If a match is not found and 802.1X authentication is enabled for the supplicant, the device continues with MAC RADIUS authentication on the authentication server.

For each MAC address in the list, you can configure the VLAN to which the supplicant is moved or the interfaces on which the supplicant can connect.

## Guest VLAN

You can specify a guest VLAN that provides limited network access for nonresponsive supplicants. If a guest VLAN is configured, the authenticator connects all nonresponsive supplicants to the predetermined VLAN, providing limited network access, often only to the Internet. This type of configuration can be used to provide Internet access to visitors without compromising company security.

NOTE: In 802.1X, MAC RADIUS, and guest VLAN must not be configured together, because guest VLAN does not work when MAC RADIUS is configured.

IEEE 802.1X provides LAN access to nonresponsive hosts, which are hosts where 802.1X is not enabled. These hosts, referred to as guests, typically are provided access only to the Internet.

## RADIUS Server Failure Fallback

You can define one of four actions to be taken if no RADIUS authentication server is reachable (if, for example, a server failure or a timeout has occurred on the authentication server).

- **deny**—(default) Prevent traffic from flowing from the supplicant through the interface.
- **permit**—Allow traffic to flow from the supplicant through the interface as if the supplicant were successfully authenticated by the RADIUS server.
- **use-cache**—Force successful authentication if authentication was granted before the failure or timeout. This ensures that authenticated users are not adversely affected by a failure or timeout.
- **vlan *vlan-name* | *vlan-id*** —Move the supplicant to a different VLAN specified by name or ID. This applies only to the first supplicant connecting to the interface.

NOTE: For the permit, use-cache, and vlan fallback actions to work, 802.1X supplicants need to accept an out-of-sequence SUCCESS packet.

For RADIUS server settings, see Table 34 on page 252.

### Table 34: RADIUS Server Settings

| Field | Function | Your Action |
|---|---|---|
| IP Address | Specifies the IP address of the server. | Enter the IP address in dotted decimal notation. |
| Password | Specifies the login password. | Enter the password. |
| Confirm Password | Verifies the login password for the server. | Reenter the password. |

Table 34: RADIUS Server Settings *(continued)*

| Field | Function | Your Action |
|---|---|---|
| Server Port Number | Specifies the port with which the server is associated. | Type the port number. |
| Source Address | Specifies the source address of the SRX Series device for communicating with the server. | Type the IP address in dotted decimal notation. |
| Retry Attempts | Specifies the number of login retries allowed after a login failure. | Type the number. |
| Timeout | Specifies the time interval to wait before the connection to the server is closed. | Type the interval in seconds. |

For 802.1X exclusion list details, see Table 35 on page 253.

Table 35: 802.1X Exclusion List

| Field | Function | Your Action |
|---|---|---|
| MAC Address | Specifies the MAC address to be excluded from 802.1X authentication. | Enter the MAC address. |
| Exclude if connected through the port | Specifies that a supplicant can bypass authentication if it is connected through a particular interface. | Select to enable the option. Select the port through which the supplicant is connected. |
| Move the host to the VLAN | Moves the host to a specific VLAN once the host is authenticated. | Select to enable the option. Select the VLAN from the list. |

For 802.1X port settings, see Table 36 on page 253.

Table 36: 802.1X Port Settings

| Field | Function | Your Action |
|---|---|---|
| **Supplicant Mode** | | |
| Supplicant Mode | Specifies the mode to be adopted for supplicants:<br>• Single secure—Allows only one host for authentication.<br>• Multiple—Allows multiple hosts for authentication. Each host is checked before being admitted to the network.<br>• Single mode authentication for multiple hosts—Allows multiple hosts but only the first is authenticated. | Select the required mode. |
| **Authentication** | | |
| Enable re-authentication | Specifies enabling reauthentication on the selected interface. | Select to enable reauthentication. Enter the timeout for reauthentication in seconds. |

Table 36: 802.1X Port Settings *(continued)*

| Field | Function | Your Action |
|---|---|---|
| Action for nonresponsive hosts | Specifies the action to be taken in case a supplicant is nonresponsive:<br>• Move to the Guest VLAN—Moves the supplicant to the specified Guest VLAN.<br>• Deny—Does not permit access to the supplicant. | Select the required action. |
| **Timeouts** | Specifies timeout values for:<br>• Port waiting time after an authentication failure<br>• EAPOL retransmitting interval<br>• Maximum EAPOL requests<br>• Maximum number of retries<br>• Port timeout value for a response from the supplicant<br>• Port timeout value for a response from the RADIUS server | Enter timeout values in seconds for the appropriate options. |

## VoIP VLAN Support

When VoIP is used with 802.1X, the RADIUS server authenticates the phone, and Link Layer Discovery Protocol–Media Endpoint Discovery (LLDP-MED) provides the class-of-service (CoS) parameters for the phone.

You can configure 802.1X authentication to work with VoIP in multiple-supplicant or single-supplicant mode:

- **Multiple-supplicant mode**—Allows multiple supplicants to connect to the interface. Each supplicant is authenticated individually.
- **Single-supplicant mode**—Authenticates only the first supplicant. All other supplicants that connect later to the interface are allowed to *piggyback* on the first supplicant's authentication and gain full access.

## RADIUS Accounting

Configuring RADIUS accounting on a SRX Series device lets you collect statistical data about users logging in to and out off a LAN, and sends it to a RADIUS accounting server. The collected data can be used for general network monitoring, to analyze and track usage patterns, or to bill a user on the basis of the amount of time or type of services accessed.

To configure RADIUS accounting, specify one or more RADIUS accounting servers to receive the statistical data from the device, and select the type of accounting data to be collected. To view the collected statistics, you can access the log file configured to receive them.

## Server Reject VLAN

By default, when authentication fails, the supplicant is denied access to the network. However, you can specify a VLAN to which the supplicant is moved if authentication fails. The server reject VLAN is similar to a guest VLAN. With a server reject VLAN, however,

authentication is first attempted by credential, then by MAC address. If both authentication methods fail, the supplicant is given access to a predetermined VLAN with limited network access.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D80 | Starting in Junos OS 15.1X49-D80, 802.1X port-based authentication is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices. |
| 15.1X49-D40 | From Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D75 and Junos OS Release 17.3R1, IEEE 802.1X port-based network authentication is not supported. |

**Related Documentation**

- *Example: Configuring 802.1x Authentication*
- Ethernet Ports Switching Overview for Security Devices on page 104
- *Verifying Switching Mode Configuration*

## Example: Specifying RADIUS Server Connections on a Security Device

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

This example shows how to specify a RADIUS server for 802.1X authentication to provide network edge security.

NOTE: From Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D75 and Junos OS Release 17.3R1, specifying a RADIUS server for 802.1X authentication is not supported.

NOTE: Starting in Junos OS 15.1X49-D80, 802.1X port-based authentication is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.



### Requirements

Before you begin, verify that the interfaces used are in switch mode. See "Example: Configuring Switching Modes on Security Devices" on page 111.

- To use 802.1X or MAC RADIUS authentication, you must specify the connections on the SRX Series device for each RADIUS server to which you will connect.

## Overview

In this example, you set the RADIUS server IP address to 10.204.96.165 and the secret password to abc. The secret password on the device must match the secret password on the server. You can set the number of retries after which port is placed into wait state to 5.

Then you create a profile called **profile1** and set the authentication order to **radius**. You can specify one or more RADIUS servers to be associated with **profile1**. Finally, you define profile1 as the authentication profile for 802.1X or MAC RADIUS authenticator.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set access radius-server 10.204.96.165 secret abc
set access radius-server 10.204.96.165 retry 5
set access profile profile1 authentication-order radius
set access profile profile1 radius authentication-server 10.204.96.165
set access profile profile1 radius accounting-server 10.204.96.165
set protocols dot1x authenticator authentication-profile-name profile1
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To specify a RADIUS server for 802.1X authentication:

1. Configure access.

   ```
   [edit]
   user@host# edit access
   ```

   NOTE: For 802.1X authentication, the RADIUS server must be configured at the access hierarchy level.

2. Define the IP address and the secret password for the RADIUS server.

   ```
   [edit access]
   user@host# set access radius-server 10.204.96.165 secret abc
   ```

3. Specify the number of retries after which port is placed into wait state to 5.

   ```
   [edit access]
   ```

```
user@host# set access radius-server 10.204.96.165 retry 5
```

4. Create the profile.

```
[edit access]
user@host# edit profile profile1
```

5. Configure the authentication order.

```
[edit access profile profile1]
user@host# set authentication-order radius
```

6. Specify one or more RADIUS servers to be associated with profile1.

```
[edit access profile profile1]
user@host# set radius authentication-server 10.204.96.165
user@host# set radius accounting-server 10.204.96.165
```

7. Define authentication profile.

```
[edit]
user@host# set protocols dot1x authenticator authentication-profile-name profile1
```

**Results** From configuration mode, confirm your configuration by entering the **show access** and **show protocols dot1x** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show access
  radius-server {
    10.204.96.165 {
      secret "$ABC123"; ## SECRET-DATA
      retry 5;
    }
  }
  profile profile1 {
    authentication-order radius;
    radius {
    authentication-server 10.204.96.165;
    accounting-server 10.204.96.165;
  }
}
[edit]
  user@host# show protocols dot1x
  authenticator {
    interface {
      ge-0/0/0.0 {
        supplicant multiple
        mac-radius;
        no-reauthentication;
        server-fail permit;
      }
      ge-0/0/1.0 {
        supplicant multiple
```

```
            mac-radius;
            no-reauthentication;
        }
    }
}
```

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

## Verification

### Verifying a RADIUS Server

**Purpose** Verify that the RADIUS server is configured properly.

**Action** From configuration mode, enter the **show access** and **show protocols dot1x** commands.

```
user@host# show access
  radius-server {
    10.204.96.165 {
      secret "$ABC123"; ## SECRET-DATA
      retry 5;
    }
  }
  profile profile1 {
    authentication-order radius;
    radius {
    authentication-server 10.204.96.165;
    accounting-server 10.204.96.165;
  }
}
user@host# show protocols dot1x
authenticator {
    static {
      00:50:56:85:66:0f/48 {
        vlan-assignment vlan6;
        interface ge-0/0/0.0;
      }
      00:50:56:9e:56:42/48 {
        vlan-assignment vlan6;
        interface ge-0/0/1.0;
      }
  }
    interface {
      ge-0/0/0.0 {
        supplicant multiple;
        server-fail deny;
      }
      ge-0/0/1.0 {
        supplicant multiple;
        server-fail deny;
      }
  }
l2-learning {
  global-mac-table-aging-time 60;
```

```
    global-mode switching;
}
```

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D80 | Starting in Junos OS 15.1X49-D80, 802.1X port-based authentication is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices. |
| 15.1X49-D40 | From Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D75 and Junos OS Release 17.3R1, specifying a RADIUS server for 802.1X authentication is not supported. |

**Related Documentation**

- *Example: Configuring 802.1x Authentication*
- Ethernet Ports Switching Overview for Security Devices on page 104
- Understanding 802.1X Port-Based Network Authentication on page 249
- Understanding Switching Modes on Security Devices on page 103
- Understanding VLANs on page 115

## Example: Configuring 802.1X Interface Settings on a Security Device

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

This example shows how to configure 802.1X interface settings for network edge security.

NOTE: From Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D75 and Junos OS Release 17.3R1, configuring 802.1X port-based authentication interface settings is not supported.

NOTE: Starting in Junos OS 15.1X49-D80, 802.1X port-based authentication is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.



### Requirements

Before you begin:

- Verify that the interfaces used are in switch mode. See "Example: Configuring Switching Modes on Security Devices" on page 111.
- Ensure that the interfaces are defined in the interfaces hierarchy with family ethernet-switching.

## Overview

In this example, you set the supplicant mode to **multiple** after configuring protocol **dot1x** and authenticator interface ge-0/0/0.0. You then enable reauthentication and set the reauthentication interval to **120**. You then configure the timeout for the interface before it resends an authentication request to the RADIUS server as **5**. You specify the time, in seconds, the interface waits before retransmitting the initial EAPoL PDUs to the supplicant as **60**. Set the **server-fail** to **deny** so that the server does not fail. Finally, you configure the maximum number of times an EAPoL request packet is retransmitted to the supplicant before the authentication session times out as **5**.

## 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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set protocols dot1x authenticator interface ge-0/0/0.0 supplicant multiple
  reauthentication 120
set protocols dot1x authenticator interface ge-0/0/0.0 server-timeout 5 transmit-period
  60
set protocols dot1x authenticator interface ge-0/0/0.0 server-fail deny
set protocols dot1x authenticator interface ge-0/0/0.0 maximum-requests 5
set protocols dot1x authenticator interface ge-0/0/2.0 supplicant multiple
  reauthentication 120
set protocols dot1x authenticator interface ge-0/0/2.0 server-timeout 5 transmit-period
  60
set protocols dot1x authenticator interface ge-0/0/2.0 server-fail deny
set protocols dot1x authenticator interface ge-0/0/2.0 maximum-requests 5
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To specify a RADIUS server for 802.1X authentication:

1. Configure the protocol.

   ```
   [edit]
   user@host# set protocols dot1x
   ```

2. Configure an interface.

   ```
   [edit protocols dot1x]
   user@host# set authenticator interface ge-0/0/0.0
   ```

3. Configure the supplicant mode.

   ```
   [edit protocols dot1x authenticator interface ge-0/0/0.0]
   user@host# set supplicant multiple
   ```

4. Enable reauthentication and specify the reauthentication interval.

   ```
   [edit protocols dot1x authenticator interface ge-0/0/0.0]
   user@host# set reauthentication 120
   ```

5. Configure and set the server timeout value.

   ```
   [edit protocols dot1x authenticator interface ge-0/0/0.0]
   user@host# set server-timeout 5
   ```

6. Configure transmit period.

   ```
   [edit protocols dot1x authenticator interface ge-0/0/0.0]
   user@host# set transmit-period 60
   ```

7. Set **server-fail** to **deny**

   ```
   [edit protocols dot1x authenticator interface ge-0/0/0.0]
   user@host# set server-fail deny
   ```

8. Specify the maximum requests value.

   ```
   [edit protocols dot1x authenticator interface ge-0/0/0.0]
   user@host# set maximum-requests 5
   ```

**Results** From configuration mode, confirm your configuration by entering the **show protocols dot1x** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show protocols dot1x
  authenticator {
    interface {
      ge-0/0/0.0 {
        supplicant multiple;
        transmit-period 60;
        mac-radius;
        reauthentication 120;
        server-timeout 5;
        maximum-requests 5;
        server-fail permit;
      }
      ge-0/0/1.0 {
        supplicant multiple;
        mac-radius;
        no-reauthentication;
      }
  }
}
```

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

## Verification

### Verifying 802.1X Interface Settings

**Purpose** Verify that the 802.1X interface settings are working properly.

**Action** From configuration mode, enter the **show protocols dot1x** command.

```
user@host# show protocols dot1x
authenticator {
    interface {
      ge-0/0/0.0 {
        supplicant multiple;
        server-fail deny;
      }
      ge-0/0/1.0 {
        supplicant multiple;
        server-fail deny;
      }
  }
```

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D80 | Starting in Junos OS 15.1X49-D80, 802.1X port-based authentication is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices. |
| 15.1X49-D40 | From Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D75 and Junos OS Release 17.3R1, configuring 802.1X port-based authentication interface settings is not supported. |

**Related Documentation**

- *Example: Configuring 802.1x Authentication*
- Ethernet Ports Switching Overview for Security Devices on page 104
- Understanding 802.1X Port-Based Network Authentication on page 249
- Understanding Switching Modes on Security Devices on page 103
- Understanding VLANs on page 115

CHAPTER 21

# Configuring Port Security



## Port Security Overview

**Supported Platforms** SRX Series

Ethernet LANs are vulnerable to attacks such as address spoofing (forging) and Layer 2 denial of service (DoS) attacks on network devices. Port security features help protect the access ports on your services gateway against the losses of information and productivity that can result from such attacks.

Junos OS on SRX Series devices provides features to help secure ports on a switching port on the services gateway. The ports can be categorized as either trusted or untrusted. You apply policies appropriate to those categories to protect against various types of attacks.

The MAC limit port security feature can be turned on to obtain the most robust port security level. Basic port security features are enabled in the services gateway's default configuration. You can configure additional features with minimal configuration steps.

**Related Documentation**

- Ethernet Ports Switching Overview for Security Devices on page 104
- Understanding MAC Limiting on page 263
- *Verifying Switching Mode Configuration*

## Understanding MAC Limiting

**Supported Platforms** SRX Series

MAC limiting protects against flooding of the Ethernet switching table (also known as the MAC forwarding table or Layer 2 forwarding table). You enable this feature on interfaces (ports).

MAC limiting sets a limit on the number of MAC addresses that can be learned dynamically on a single Layer 2 access interface or on all the Layer 2 access interfaces on the services gateway.

You configure the maximum number of dynamic MAC addresses allowed per interface. When the limit is exceeded, incoming packets with new MAC addresses are treated as specified by the configuration.

You can choose to have one of the following actions performed when the MAC addresses limit is exceeded:

NOTE: Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, the **log**, **none**, and **shutdown** actions are not supported.

- **drop**—Drop the packet and generate an alarm, an SNMP trap, or a system log entry. This is the default.
- **log**—Do not drop the packet but generate an alarm, an SNMP trap, or a system log entry.
- **none**—Take no action.
- **shutdown**—Disable the interface and generate an alarm. If you have configured the services gateway with the **port-error-disable** statement, the disabled interface recovers automatically upon expiration of the specified disable timeout. If you have not configured the services gateway for autorecovery from port error disabled conditions, you can bring up the disabled interfaces with running the **clear ethernet-switching recovery-timeout** command.

NOTE: MAC limit is applied only to new MAC learning requests. If you already have 10 learned MAC addresses and you configure the limit as 5, all the MACs will remain in the forwarding database (FDB) table. When the learned MAC addresses age out (or are cleared by the user with the clear ethernet-switching command), they are not relearned.

MAC limiting does not apply to static MAC addresses. Users can configure any number of static MAC addresses independent of MAC limiting and all of them are added to FDB.

NOTE: Starting with Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, the maximum number of MAC addresses learned on all logical interfaces on the SRX1500 device is 24,575. When this limit is reached, incoming packets with a new source MAC address will be dropped.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D40 | Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, the **log**, **none**, and **shutdown** actions are not supported. |

**Related Documentation**

- Example: Configuring MAC Limiting on a Security Device on page 265
- Port Security Overview on page 263
- Ethernet Ports Switching Overview for Security Devices on page 104
- *Verifying Switching Mode Configuration*

## Example: Configuring MAC Limiting on a Security Device

**Supported Platforms** SRX Series

This example shows how to configure port security features by setting a MAC limit of 5.



### Requirements

Before you begin, verify that the interfaces that will be used are in switch mode. See "Example: Configuring Switching Modes on Security Devices" on page 111 and "Understanding Switching Modes on Security Devices" on page 103.

### Overview

MAC limiting protects against flooding of the Ethernet switching table on the SRX Series Services Gateways. MAC limiting sets a limit on the number of MAC addresses that can be learned on a single Layer 2 access interface (port).

This example shows how to configure port security features by setting a MAC limit of 5.

**CLI Quick Configuration**

To quickly configure this example, copy the following command, paste it into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the command into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
[edit]
set switch-options interface ge-0/0/1 interface-mac-limit 5
set interface ge–0/0/2 ether-options source-address-filter 00:00:5E:00:AA
set interface ge–0/0/2 ether-options source-address-filter 00:00:5E:00:AB
set interface ge–0/0/2 ether-options source-address-filter 00:00:5E:00:AC
```

## Configuration

**Step-by-Step Procedure**

The action is not specified, so that the device performs the default action **drop** if the limit is exceeded:

NOTE: Do not set the mac-limit to 1. The first learned MAC address is often inserted into the FDB automatically (for example, for routed VLAN interfaces the first MAC address inserted into the forwarding database is the MAC address of the RVI; for Aggregated Ethernet bundles using LACP, the first MAC address inserted into the FDB in the forwarding table is the source address of the protocol packet). The services gateway will therefore not learn MAC addresses other than the automatic addresses when the mac-limit is set to 1, and this will cause problems with MAC learning and forwarding.

1. On a single interface (here, the interface is ge-0/0/1):

```
[edit switch-options]
user@host# set switch-options interface ge-0/0/1 interface-mac-limit 5
```

2. For specifying specific MAC addresses:

   - On a single interface (here, the interface is ge-0/0/2):

```
[edit interfaces ether-options source-address-filter ]
user@host# set interface ge–0/0/2 ether-options source-address-filter
   00:00:5E:00:AA
user@host# set interface ge–0/0/2 ether-options source-address-filter
   00:00:5E:00:AB
user@host# set interface ge–0/0/2 ether-options source-address-filter
   00:00:5E:00:AC
```

3. Enter **commit** from configuration mode.

## Verification

### Verifying That MAC Limiting Is Working Correctly on the Services Gateway

**Purpose** Verify that MAC limiting is working on the services gateway.

**Action** Display the learned MAC addresses. The following sample output shows the results when two packets were sent from hosts on ge-0/0/1 and five packets requests were sent from hosts on ge-0/0/2, with both interfaces set to a MAC limit of 4 with the action drop:

```
user@host> show ethernet-switching table
Ethernet-switching table: 7 entries, 6 learned
VLAN MAC address Type Age Interfaces
employee-vlan * Flood - ge-0/0/2.0
employee-vlan 00:00:5E:00:00 Learn 0 ge-0/0/1.0
employee-vlan 00:00:5E:00:AA Learn 0 ge-0/0/1.0
```

```
employee-vlan 00:00:5E:00:AB Learn 0 ge-0/0/2.0
employee-vlan 00:00:5E:00:AC Learn 0 ge-0/0/2.0
employee-vlan 00:00:5E:00:AD Learn 0 ge-0/0/2.0
employee-vlan 00:00:5E:00:AE Learn 0 ge-0/0/2.0
```

**Meaning** The sample output shows that with a MAC limit of 4 for each interface, the packet for a fifth MAC address on ge-0/0/2 was dropped because it exceeded the MAC limit. The address was not learned, and thus an asterisk (*) rather than an address appears in the MAC address column in the first line of the sample output.

**Related Documentation**

- Understanding MAC Limiting on page 263
- Ethernet Ports Switching Overview for Security Devices on page 104
- *Verifying Switching Mode Configuration*

## Configuring Autorecovery From the Disabled State on Secure Interfaces of a Security Device (CLI Procedure)

**Supported Platforms** SRX300, SRX320, SRX340, SRX345, SRX550M

An Ethernet switching access interface on a SRX Series device might shut down or be disabled as a result of one of the following port-security configurations:

- MAC limiting—**mac-limit** statement is configured with action **shutdown**.

You can configure the device to automatically restore the disabled interfaces to service after a specified period of time. Autorecovery applies to all the interfaces that have been disabled due to MAC limiting errors.

NOTE: You must specify the disable timeout value for the interfaces to recover automatically. There is no default disable timeout. If you do not specify a timeout value, you need to use the clear ethernet-switching recovery-timeout command to clear the errors and restore the interfaces or the specified interface to service.

To configure autorecovery from the disabled state due to MAC limiting shutdown actions:

```
[edit interfaces]
user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching recovery-timeout 60
```

**Related Documentation**

- Understanding MAC Limiting on page 263
- Example: Configuring MAC Limiting on a Security Device on page 265
- clear ethernet-switching recovery-timeout on page 394

CHAPTER 22

# Configuring Ethernet OAM Connectivity Fault Management



## Understanding Ethernet OAM Connectivity Fault Management

**Supported Platforms** SRX Series

Ethernet interfaces on SRX Series devices support the IEEE 802.1ag standard for Operation, Administration, and Management (OAM). The 802.1ag is an IEEE standard for connectivity fault management (CFM). The IEEE 802.1ag provides a specification for Ethernet CFM. The Ethernet network can consist of one or more service instances. A service instance could be a VLAN or a concatenation of VLANs. The goal of CFM is to provide a mechanism to monitor, locate, and isolate faulty links.

NOTE: **Support for the IEEE 802.1ag standard for OAM on SRX Series devices depends on the Junos OS release running on the device.**

Starting in Junos OS Release 15.1X49-D80, Ethernet OAM CFM is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, Ethernet OAM CFM is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Ethernet OAM CFM is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

CFM support includes the following features:

- Fault monitoring using the Continuity Check Protocol. This is a neighbor discovery and health check protocol that discovers and maintains adjacencies at the VLAN or link level.
- Path discovery and fault verification using the Link Trace protocol. This feature is not supported in Junos OS Release 12.3X48-D65.
- Fault isolation using the Loopback protocol.

  The Loopback protocol is used to check access to maintenance association end points (MEPs) under the same maintenance association (MA). The Loopback messages are triggered by an administrator using the **ping ethernet** command.

NOTE: Virtual private LAN service (VPLS) is not supported on SRX100, SRX110, SRX210, SRX220, SRX240, SRX300, SRX320, SRX340, SRX345, SRX550M, SRX1400, and SRX1500 devices.

CFM partitions the service network into various administrative domains. For example, operators, providers, and customers might be part of different administrative domains. Each administrative domain is mapped into one maintenance domain providing enough information to perform its own management, thus avoiding security breaches and making end-to-end monitoring possible.

In a CFM maintenance domain, each service instance is called a maintenance association. A maintenance association can be thought of as a full mesh of maintenance association end points (MEPs) having similar characteristics. MEPs are active CFM entities generating and responding to CFM protocol messages. There is also a maintenance association intermediate point (MIP), which is a CFM entity similar to the MEP, but more passive (MIPs only respond to CFM messages).

Each maintenance domain is associated with a maintenance domain level from 0 through 7. Level allocation is based on the network hierarchy, where outer domains are assigned a higher level than the inner domains. You configure customer end points to have the highest maintenance domain level. The maintenance domain level is a mandatory parameter that indicates the nesting relationships between various maintenance domains. The level is embedded in each CFM frame. CFM messages within a given level are processed by MEPs at that same level.

To enable CFM on an Ethernet interface, you must configure maintenance domains, maintenance associations, and MEPs.

The limitations for CFM are as follows:

- You cannot configure MEP and MIP on the same VLAN.
- CFM and link fault management (LFM) can be configured on the same interface.

- You cannot configure CFM with Generic VLAN Registration Protocol (GVRP).
- CFM is not supported on VoIP VLAN ports.
- On SRX240, and SRX550M devices, the default Loopback message (LBM) packet size is 113 bytes.

## Benefits of Ethernet CFM

Ethernet CFM provides the following benefits:

- End-to-end service-level OAM technology
- Reduced operating expense for service provider Ethernet networks
- Competitive advantage for service providers

## CFM over VDSL and PPPoE interfaces for SRX210, SRX220, SRX240, SRX320, SRX340, SRX345, SRX550, and SRX550M Devices

Starting in Junos OS Release 12.3X48-D65, on SRX210, SRX220, SRX240, and SRX550 devices, Operation, Administration, and Maintenance (OAM) connectivity fault management (CFM) is supported on very-high-bit-rate digital subscriber line (VDSL) and Point-to-Point Protocol over Ethernet (PPPoE) interfaces in addition to Ethernet interfaces.

CFM over VDSL should be configured on the pt interface. To support CFM over PPPoE, you need to configure maintenance domain and maintenance association end point (MEP). The CFM over VDSL interface supports down direction MEP, continuity check, and loopback protocols.

The following are the limitations when configuring Ethernet CFM over VDSL or Layer 3 interface:

- CFM action profiles are not supported on the Point-to-Point Protocol over Ethernet (PPPoE) logical interface on SRX210, SRX220, SRX240, SRX550, and SRX650 devices.
- Synthetic loss measurement on demand is supported only on SRX320, SRX340, SRX345, and SRX550M devices. Proactive synthetic loss measurement is not supported.
- When CFM over PPPoE is implemented, CFM must be applied on the PPPoE logical interface and not on the underlying interface.
- CFM over VDSL can be implemented as a MEP but not as a MIP.
- CFM higher-level pass-through over a VDSL or Gigabit Ethernet interface in Layer 3 interface mode is not supported.
- For a VLAN-tagged VDSL interface, CFM must always be applied on the respective logical interface and not over the physical interface.
- When CFM is enabled on VDSL, CFM packets are dropped randomly, causing CFM sessions to flap based on the timer when transit traffic exceeds the line rate. Flapping occurs because the VDSL Mini-Physical Interface Module (Mini-PIM) cannot differentiate and prioritize CFM packets.

**Related Documentation**

- Example: Configuring Ethernet OAM Connectivity Fault Management on a Security Device on page 272

## Example: Configuring Ethernet OAM Connectivity Fault Management on a Security Device

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D80, Ethernet OAM connectivity fault management is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, Ethernet OAM connectivity fault management is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Ethernet OAM connectivity fault management is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

Connectivity Fault Management (CFM) provides a mechanism to monitor, locate, and isolate faulty links.

This example describes how to enable and configure an end-to-end OAM CFM session on an Ethernet interface.



### Requirements

This example uses the following hardware and software components:

- Three SRX Series devices connected by a point-to-point Ethernet link.
- Junos OS Release 12.1X44-D10 or later for SRX Series devices.

### Overview

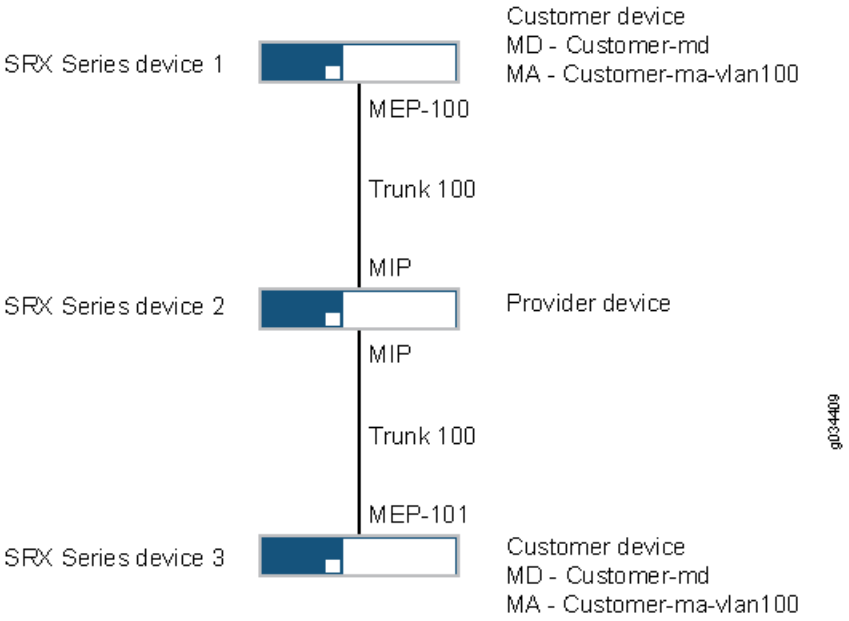
Ethernet interfaces on SRX Series devices support the IEEE 802.1ag standard for Operation, Administration, and Management (OAM). The IEEE 802.1ag specification provides a specification for Ethernet connectivity fault management (CFM). CFM can be used to detect faults in the network path between the customer premises devices. It also helps in detecting the device or node in the provider network, where the failure occurred.

This example describes how to configure an end to end CFM session. In this example, three devices are connected by a point-to-point Ethernet link. The link between these devices is monitored using CFM. To check connectivity or fault through the provider network, maintenance intermediate point (MIP) is configured.

### Topology

Figure 11 on page 273 shows three SRX Series devices connected by a point-to-point Ethernet link.

**Figure 11: Ethernet CFM with SRX Series Devices**




**Legend**
MA - Maintenance Association
MD - Maintenance Domain
MEP - Maintenance Association End Point
MIP - Maintenance Association Intermediate Point

## Configuring Ethernet OAM Connectivity Fault Management



### Configuring Ethernet OAM Connectivity Fault Management on Device 1

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/4 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/4 unit 0 family ethernet-switching vlan members v100
set vlans v100 vlan-id 100
```

```
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md level 5
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 100 interface ge-0/0/4.0
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 100 interface vlan 100
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 100 auto-discovery
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma continuity-check interval 10s
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma continuity-check hold-interval
  20
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To enable and configure OAM CFM on device 1:

1. Define a VLAN and enable the interface for family Ethernet switching with interface mode trunk or access.

   ```
   [edit]
   user@host# set interfaces ge-0/0/4 unit 0 family ethernet-switching interface-mode
     trunk
   user@host# set interfaces ge-0/0/4 unit 0 family ethernet-switching vlan members
     v100
   user@host# set vlans v100 vlan-id 100
   ```

2. Specify the maintenance domain name and the maintenance domain level.

   ```
   [edit protocols oam ethernet connectivity-fault-management ]
   user@host# set maintenance-domain Customer-md level 5
   ```

3. Create a maintenance association and configure MEP.

   ```
   [edit protocols oam ethernet connectivity-fault-management maintenance-domain
     Customer-md]
   user@host# set maintenance-association Customer-ma mep 100 interface
     ge-0/0/4.0
   user@host# set maintenance-association Customer-ma mep 100 interface vlan
     100
   ```

4. Enable MEP automatic discovery.

   ```
   [edit protocols oam ethernet connectivity-fault-management maintenance-domain
     Customer-md maintenance-association Customer-ma]
   user@host# set mep 100 auto-discovery
   ```

5. Enable the Continuity Check Protocol and specify the continuity check interval and hold interval.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma]
user@host# set continuity-check interval 10s
user@host# set continuity-check hold-interval 20
```

**Results** From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

For brevity, this **show protocols** command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
[edit]
user@host# show protocols

oam {
  ethernet {
    connectivity-fault-management {
      maintenance-domain Customer-md {
        level 5;
        maintenance-association Customer-ma {
          continuity-check {
            interval 10s;
            hold-interval 20;
          }
          mep 100 {
            interface ge-0/0/4.0 vlan 100;
            auto-discovery;
          }
        }
      }
    }
  }
}
```

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

## Configuring Ethernet OAM CFM with MIP Half Function on Device 2

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members v100
set interfaces ge-0/0/4 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/4 unit 0 family ethernet-switching vlan members v100
set vlans v100 vlan-id 100
set protocols oam ethernet connectivity-fault-management maintenance-domain
  default-5 v100
```

```
set protocols oam ethernet connectivity-fault-management maintenance-domain
  default-5 mip-half-function default
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure MIP half function:

1. Define a VLAN and enable the interface for family Ethernet switching with interface mode trunk or access.

```
[edit]
user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode
  trunk
user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members
  v100
user@host# set interfaces ge-0/0/4 unit 0 family ethernet-switching interface-mode
  trunk
user@host# set interfaces ge-0/0/4 unit 0 family ethernet-switching vlan members
  v100
user@host# set vlans v100 vlan-id 100
```

2. Create a maintenance domain and configure VLAN.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenance-domain default-5 v100
```

3. Create a MIP half function.

```
[edit protocols oam ethernet connectivity-fault-management ]
user@host# set maintenance-domain default-5 mip-half-function default
```

NOTE: If you want to configure traceoptions, run the following commands:

```
set protocols oam ethernet connectivity-fault-management traceoptions
  file CFM_trace
set protocols oam ethernet connectivity-fault-management traceoptions
  flag all
```

**Results**

From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

For brevity, this **show** command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
[edit]
```

```
user@host# show protocols
  oam {
    ethernet {
      connectivity-fault-management {
        traceoptions {
          file CFM_trace;
          flag all;
        }
        maintenance-domain default-5 {
          v100;
          mip-half-function default;
        }
      }
    }
  }
```

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

## Configuring Ethernet OAM Connectivity Fault Management on Device 3

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members v100
set vlans v100 vlan-id 100
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md level 5
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 101 interface ge-0/0/1.0
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 101 interface vlan 100
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 101 auto-discovery
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma continuity-check hold-interval
  20
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma continuity-check interval 10s
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To enable and configure OAM CFM on Device 3:

1. Define a VLAN and enable the interface for family Ethernet switching with interface mode trunk or access.

   ```
   [edit]
   user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching interface-mode
     trunk
   ```

```
user@host# set interfaces ge-0/0/1 unit 0 family ethernet-switching vlan members
  v100
user@host# set vlans v100 vlan-id 100
```

2. Specify the maintenance domain name and the maintenance domain level.

```
[edit protocols oam ethernet connectivity-fault-management ]
user@host# set maintenance-domain Customer-md level 5
```

3. Create a maintenance association and configure MEP.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md]
user@host# set maintenance-association Customer-ma mep 101 interface
  ge-0/0/1.0
user@host# set maintenance-association Customer-ma mep 101 interface vlan 100
```

4. Enable MEP automatic discovery.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md]
user@host# set maintenance-association Customer-ma mep 101 auto-discovery
```

5. Enable the Continuity Check Protocol and specify the continuity check interval and hold interval.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma]
user@host# set continuity-check interval 10s
user@host# set continuity-check hold-interval 20
```

**Results** From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

For brevity, this **show** command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
[edit]
user@host# show protocols
  oam {
    ethernet {
      connectivity-fault-management {
        maintenance-domain Customer-md {
          level 5;
          maintenance-association Customer-ma {
            continuity-check {
              interval 10s;
              hold-interval 20;
            }
            mep 101 {
              interface ge-0/0/1.0 vlan 100;
```

```
            auto-discovery;
          }
        }
      }
    }
  }
}
```

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

## Verification

Confirm that the configuration is working properly.


- Verifying the OAM CFM Configuration on Device 1 on page 279
- Verifying the OAM CFM Configuration with MIP Half Function on Device 2 on page 280
- Verifying the OAM CFM Configuration on Device 3 on page 281
- Verifying the Path Using the Link Trace Protocol on page 282
- Verifying MEP Continuity Using Ping on page 282


### Verifying the OAM CFM Configuration on Device 1

**Purpose** Verify that OAM CFM has been configured properly.

**Action** From operational mode, enter the following commands:

- **show oam ethernet connectivity-fault-management adjacencies** to display connectivity-fault-management adjacencies.
- **show oam ethernet connectivity-fault-management interfaces** to display the Ethernet OAM information for the specified interface.

These commands produce the following sample output:

```
user@host# show oam ethernet connectivity-fault-management adjacencies
Mep-id   Interface          State              Timer to Expire
          101     ge-0/0/4.0         ok                 29

user@host# show oam ethernet connectivity-fault-management interfaces
Interface         Link      Status       Level  MEP         Neighbours
                                                           Identifier
       ge-0/0/4.0          Up        Active       5      100         1

user@host# show oam ethernet connectivity-fault-management interfaces detail
Interface name: ge-0/0/4.0, vlan 100, Interface status: Active, Link status: Up
  Maintenance domain name: Customer-md, Format: string, Level: 5
    Maintenance association name: Customer-ma, Format: string
    Continuity-check status: enabled, Interval: 10s
    MEP identifier: 100, Direction: down, MAC address: 2c:6b:f5:62:29:84
    MEP status: running
    Defects:
      Remote MEP not receiving CCM                : no
      Erroneous CCM received                      : no
      Cross-connect CCM received                  : no
```

```
    RDI sent by some MEP                          : no
  Statistics:
    CCMs sent                                     : 7
    CCMs received out of sequence                 : 0
    LBMs sent                                     : 0
    Valid in-order LBRs received                  : 0
    Valid out-of-order LBRs received              : 0
    LBRs received with corrupted data             : 0
    LBRs sent                                     : 0
    LTMs sent                                     : 0
    LTMs received                                 : 0
    LTRs sent                                     : 0
    LTRs received                                 : 0
    Sequence number of next LTM request           : 0
    1DMs sent                                     : 0
    Valid 1DMs received                           : 0
    Invalid 1DMs received                         : 0
    DMMs sent                                     : 0
    DMRs sent                                     : 0
    Valid DMRs received                           : 0
    Invalid DMRs received                         : 0
  Remote MEP count: 1
    Identifier    MAC address        State    Interface
      101     80:71:1f:ad:53:81       ok    ge-0/0/4.0
```

**Meaning**

- If the **show oam ethernet connectivity-fault-management interfaces detail** command output displays continuity-check status as **enabled** and displays details of the remote MEP, it means that connectivity fault management (CFM) was configured properly.
- If the **show oam ethernet connectivity-fault-management adjacencies** command output displays the state as **ok**, it indicates that the Continuity Check Protocol is up.

### Verifying the OAM CFM Configuration with MIP Half Function on Device 2

**Purpose** Verify that OAM CFM has been configured properly.

**Action** From operational mode, run the **show oam ethernet connectivity-fault-management mip** command.

```
user@host# show oam ethernet connectivity-fault-management mip vlan 100
default maintenance-domain mhf    : default

  Interface       Level
  ge-0/0/1.0          5
  ge-0/0/4.0          5
```

**Meaning** The **show oam ethernet connectivity-fault-management mip** command output displays the MIP information.

### Verifying the OAM CFM Configuration on Device 3

**Purpose** Verify that OAM CFM has been configured properly.

**Action** From operational mode, enter the following commands:

- **show oam ethernet connectivity-fault-management adjacencies** to display connectivity-fault-management adjacencies.
- **show oam ethernet connectivity-fault-management interfaces** to display the Ethernet OAM information for the specified interface.

```
user@host# show oam ethernet connectivity-fault-management adjacencies
Mep-id      Interface          State       Timer to Expire
          100       ge-0/0/1.0         ok            27

user@host# show oam ethernet connectivity-fault-management interfaces detail
Interface name: ge-0/0/1.0, vlan 100, Interface status: Active, Link status: Up
  Maintenance domain name: Customer-md, Format: string, Level: 5
    Maintenance association name: Customer-ma, Format: string
    Continuity-check status: enabled, Interval: 10s
    MEP identifier: 101, Direction: down, MAC address: 80:71:1f:ad:53:81
    MEP status: running
    Defects:
      Remote MEP not receiving CCM                 : no
      Erroneous CCM received                       : no
      Cross-connect CCM received                   : no
      RDI sent by some MEP                         : no
    Statistics:
      CCMs sent                                    : 77
      CCMs received out of sequence                : 0
      LBMs sent                                    : 0
      Valid in-order LBRs received                 : 0
      Valid out-of-order LBRs received             : 0
      LBRs received with corrupted data            : 0
      LBRs sent                                    : 0
      LTMs sent                                    : 0
      LTMs received                                : 0
      LTRs sent                                    : 0
      LTRs received                                : 0
      Sequence number of next LTM request          : 0
      1DMs sent                                    : 0
      Valid 1DMs received                          : 0
      Invalid 1DMs received                        : 0
      DMMs sent                                    : 0
      DMRs sent                                    : 0
      Valid DMRs received                          : 0
      Invalid DMRs received                        : 0
    Remote MEP count: 1
      Identifier    MAC address        State    Interface
        100     2c:6b:f5:62:29:84       ok    ge-0/0/1.0
```

**Meaning**

- If the **show oam ethernet connectivity-fault-management interfaces detail** command output displays continuity-check status as **enabled** and displays details of the remote MEP, it means that connectivity fault management (CFM) was configured properly.

- If the **show oam ethernet connectivity-fault-management adjacencies** command output displays the state as **ok**, it indicates that the Continuity Check Protocol is up.

### Verifying the Path Using the Link Trace Protocol

**Purpose** Verify the path between maintenance endpoints.

**Action** From operational mode, enter the **traceroute ethernet** command.

```
user@host# traceroute ethernet maintenance-domain Customer-md maintenance-association
Customer-ma mep 101
Linktrace to 80:71:1f:ad:53:81, Interface : ge-0/0/4.0
      Maintenance Domain: Customer-md, Level: 5
      Maintenance Association: Customer-ma, Local Mep: 100
      Transaction Identifier: 3
    Hop   TTL    Source MAC address        Next-hop MAC address
    .
    1     63     80:71:1f:ad:50:01         80:71:1f:ad:50:01
    2     62     80:71:1f:ad:53:81         00:00:00:00:00:00
```

### Verifying MEP Continuity Using Ping

**Purpose** Verify access to MEPs under the same maintenance association.

**Action** From operational mode, enter the **ping ethernet** command.

```
user@host# ping ethernet maintenance-domain Customer-md maintenance-association
Customer-ma mep 101
PING to 80:71:1f:ad:53:81, Interface ge-0/0/4.0
60 bytes from 80:71:1f:ad:53:81: lbm_seq=0
60 bytes from 80:71:1f:ad:53:81: lbm_seq=1
60 bytes from 80:71:1f:ad:53:81: lbm_seq=2
60 bytes from 80:71:1f:ad:53:81: lbm_seq=3
--- ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
```

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269

## Example: Configuring Ethernet OAM Connectivity Fault Management over VDSL Interface

**Supported Platforms** SRX Series

Starting with 12.3X48-D65, Operation, Administration, and Maintenance (OAM) connectivity fault management (CFM) is supported on VDSL interface on SRX210, SRX220, SRX240, and SRX550 devices.

This example describes how to enable and configure an end-to-end OAM CFM session on a VDSL interface.



## Requirements

This example uses the following hardware and software components:

- Two SRX Series devices connected by VDSL MEP and MEP links.
- Junos OS Release 12.3X48-D65 or later.

## Overview

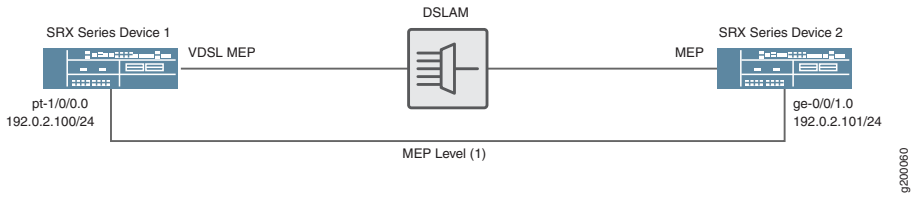
CFM can be used to detect faults in the network path between the customer premises devices. It also helps in detecting the device or node in the provider network, where the failure occurred.

This example describes how to configure an end-to-end CFM session. In this example, two devices are connected by a VDSL MEP and MEP links. The link between these devices is monitored using CFM. To check connectivity or fault through the provider network, DSL access multiplexer (DSLAM) is configured.

### Topology

Figure 12 on page 283 shows the configuration of CFM on two SRX Series devices connected by a VDSL MEP and MEP links to DSLAM to check connectivity or fault through the provider network.

**Figure 12: CFM on VDSL Interface**




## Configuring VDSL OAM Connectivity Fault Management



### Configuring VDSL OAM Connectivity Fault Management on Device 1

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces pt-1/0/0 vlan-tagging
set interfaces pt-1/0/0 unit 0 vlan-id 901
set interfaces pt-1/0/0 unit 0 family inet address 192.0.2.100/24
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md level 1
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma continuity-check interval 1s
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 200 interface pt-1/0/0.0
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 200 direction down
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 200 auto-discovery
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To enable and configure OAM CFM on device 1:

1. Define a VLAN and enable the interface for family inet address.

   ```
   [edit]
   user@host# set interfaces pt-1/0/0 vlan-tagging
   user@host# set interfaces pt-1/0/0 unit 0 vlan-id 901
   user@host# set interfaces pt-1/0/0 unit 0 family inet address 192.0.2.100/24
   ```

   NOTE: CFM also supports untagged VDSL interfaces.

2. Specify the maintenance domain name and the maintenance domain level.

   ```
   [edit protocols oam ethernet connectivity-fault-management]
   user@host# set maintenance-domain Customer-md level 1
   ```

3. Enable the Continuity Check Protocol and specify the continuity check interval.

   ```
   [edit protocols oam ethernet connectivity-fault-management maintenance-domain
     Customer-md maintenance-association Customer-ma]
   user@host# set continuity-check interval 1s
   ```

4. Create a maintenance association and configure MEP.

   ```
   [edit protocols oam ethernet connectivity-fault-management maintenance-domain
     Customer-md]
   user@host# set maintenance-association Customer-ma mep 200 interface
     pt-1/0/0.0
   ```

5. Specify that CFM CCM packets be transmitted only in one direction for the MEP. That is, set the direction as down so that CCMs are transmitted only out of (not into) the interface configured on this MEP.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
   Customer-md maintenance-association Customer-ma mep 200]
user@host# set direction down
```

6. Enable MEP automatic discovery.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
   Customer-md maintenance-association Customer-ma]
user@host# set mep 200 auto-discovery
```

NOTE: The **show cfm adjacency** command output does not display the neighbor MEP adjacency status, if **auto-discovery** mode for CFM is not enabled and static remote MEP is configured on local MEP.

**Results** From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show protocols

oam {
   ethernet {
      connectivity-fault-management {
         maintenance-domain Customermd {
            level 1;
            maintenance-association Customer-ma {
               continuity-check {
                  interval 1s;
               }
               mep 200 {
                  interface pt-1/0/0.0;
                  direction down;
                  auto-discovery;
               }
            }
         }
      }
   }
}
```

## Configuring Ethernet OAM Connectivity Fault Management on Device 2

**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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces ge-0/0/1 promiscuous-mode
set interfaces ge-0/0/1 vlan-tagging
set interfaces ge-0/0/1 unit 0 vlan-id 901
set interfaces ge-0/0/1 unit 0 family inet address 11.1.1.2/24
set protocols oam ethernet connectivity-fault-management traceoptions file cfmd
set protocols oam ethernet connectivity-fault-management traceoptions flag all
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md level 1
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma continuity-check interval 1s
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 201 interface ge-0/0/1.0
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 201 direction down
set protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 201 auto-discovery
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To enable and configure OAM CFM on device 2:

1. Enable promiscuous mode on the interface.

   ```
   [edit]
   user@host# set interfaces ge-0/0/1 promiscuous-mode
   ```

   NOTE: Enable promiscuous mode on a Gigabit Ethernet interface to view the CFM adjacency on Broadband Remote Access Server (B-RAS) running Junos OS Release 12.3X48.

2. Define a VLAN and enable the interface for family inet address.

   ```
   [edit]
   user@host# set interfaces ge-0/0/1 vlan-tagging
   user@host# set interfaces ge-0/0/1 unit 0 vlan-id 901
   user@host# set interfaces ge-0/0/1 unit 0 family inet address 11.1.1.2/24
   ```

3. Configure a trace options to the customer maintenance domain.

   ```
   [edit]
   user@host# set protocols oam ethernet connectivity-fault-management
     traceoptions file cfmd
   user@host# set protocols oam ethernet connectivity-fault-management
     traceoptions flag all
   ```

4. Specify the maintenance domain name and the maintenance domain level.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenance-domain Customer-md level 1
```

5. Enable the Continuity Check Protocol and specify the continuity check interval.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma]
user@host# set continuity-check interval 1s
```

6. Create a maintenance association and configure MEP.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md]
user@host# set maintenance-association Customer-ma mep 201 interface
  ge-0/0/1.0
```

7. Specify that CFM CCM packets be transmitted only in one direction for the MEP. That is, set the direction as down so that CCMs are transmitted only out of (not into) the interface configured on this MEP.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md maintenance-association Customer-ma mep 201]
user@host# set direction down
```

8. Enable MEP automatic discovery.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain
  Customer-md]
user@host# set maintenance-association Customer-ma mep 201 auto-discovery
```

**Results** From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show protocols
oam {
  ethernet {
    connectivity-fault-management {
      traceoptions {
        file cfmd;
        flag all;
      }
      maintenance-domain Customer-md {
        level 1;
        maintenance-association Customer-ma {
          continuity-check {
            interval 1s;
          }
          mep 201 {
            interface ge-0/0/1.0;
            direction down;
            auto-discovery;
```

```
            }
          }
        }
      }
    }
  }
```

## Verification

Confirm that the configuration is working properly.


- Verifying the OAM CFM Configuration on device 1 on page 288
- Verifying the OAM CFM Configuration on device 2 on page 289
- Verifying MEP Continuity Using Ping on page 290


### Verifying the OAM CFM Configuration on device 1

**Purpose** Verify that OAM CFM has been configured properly.

**Action** From operational mode, enter the following commands:

- **show oam ethernet connectivity-fault-management adjacencies** to display CFM adjacencies.
- **show oam ethernet connectivity-fault-management interfaces** to display the Ethernet OAM information for the specified interface.

These commands produce the following sample output:

```
user@host> show oam ethernet connectivity-fault-management adjacencies
Mep-id   Interface          State             Timer to Expire
  201      pt-1/0/0.0         ok                  2

user@host> show oam ethernet connectivity-fault-management interfaces
Interface         Link      Status       Level  MEP Identifier         Neighbours

 pt-1/0/0.0       Up        Active         1       200                      1

user@host> show oam ethernet connectivity-fault-management interfaces detail
Interface name: pt-1/0/0.0, Interface status: Active, Link status: Up
  Maintenance domain name: Customermd, Format: string, Level: 1
    Maintenance association name: Customer-ma, Format: string
    Continuity-check status: enabled, Interval: 1s
    MEP identifier: 200, Direction: down, MAC address: 2001:db8:2f:ca:8b:ab
    MEP status: running
    Defects:
      Remote MEP not receiving CCM                : no
      Erroneous CCM received                      : no
      Cross-connect CCM received                  : no
      RDI sent by some MEP                        : no
    Statistics:
      CCMs sent                                   : 1416
      CCMs received out of sequence               : 0
      LBMs sent                                   : 0
      Valid in-order LBRs received                : 0
```

```
    Valid out-of-order LBRs received              : 0
    LBRs received with corrupted data             : 0
    LBRs sent                                     : 0
    LTMs sent                                     : 0
    LTMs received                                 : 0
    LTRs sent                                     : 0
    LTRs received                                 : 0
    Sequence number of next LTM request           : 0
    1DMs sent                                     : 0
    Valid 1DMs received                           : 0
    Invalid 1DMs received                         : 0
    DMMs sent                                     : 0
    DMRs sent                                     : 0
    Valid DMRs received                           : 0
    Invalid DMRs received                         : 0
  Remote MEP count: 1
    Identifier    MAC address        State    Interface
       201     2001:db8:9c:a5:41:3d       ok    pt-1/0/0.0
```

**Meaning**

- If the **show oam ethernet connectivity-fault-management interfaces detail** command output displays continuity-check status as **enabled** and displays details of the MEP, it means that CFM was configured properly.
- If the **show oam ethernet connectivity-fault-management adjacencies** command output displays the state as **ok**, it indicates that the Continuity Check Protocol is up.

### Verifying the OAM CFM Configuration on device 2

**Purpose** Verify that OAM CFM has been configured properly.

**Action** From operational mode, enter the following commands:

- **show oam ethernet connectivity-fault-management adjacencies** to display CFM adjacencies.
- **show oam ethernet connectivity-fault-management interfaces detail** to display the Ethernet OAM information for the specified interface.

```
user@host> show oam ethernet connectivity-fault-management adjacencies
Mep-id      Interface          State       Timer to Expire
    200       ge-0/0/1.0         ok            3

user@host> show oam ethernet connectivity-fault-management interfaces detail
Interface name: ge-0/0/1.0, Interface status: Active, Link status: Up
  Maintenance domain name: Customermd , Format: string, Level: 1
    Maintenance association name: Customer-ma, Format: string
    Continuity-check status: enabled, Interval: 1s
    MEP identifier: 201, Direction: down, MAC address: 2001:db8:9c:a5:41:3d
    MEP status: running
    Defects:
      Remote MEP not receiving CCM                : no
      Erroneous CCM received                      : no
      Cross-connect CCM received                  : no
      RDI sent by some MEP                        : no
```

```
Statistics:
  CCMs sent                                    : 1486
  CCMs received out of sequence                : 0
  LBMs sent                                    : 0
  Valid in-order LBRs received                 : 0
  Valid out-of-order LBRs received             : 0
  LBRs received with corrupted data            : 0
  LBRs sent                                    : 0
  LTMs sent                                    : 0
  LTMs received                                : 0
  LTRs sent                                    : 0
  LTRs received                                : 0
  Sequence number of next LTM request          : 0
  1DMs sent                                    : 0
  Valid 1DMs received                          : 0
  Invalid 1DMs received                        : 0
  DMMs sent                                    : 0
  DMRs sent                                    : 0
  Valid DMRs received                          : 0
  Invalid DMRs received                        : 0
Remote MEP count: 1
  Identifier    MAC address        State    Interface
     200     2001:db8:2f:ca:8b:ab       ok    ge-0/0/1.0
```

**Meaning**

- If the **show oam ethernet connectivity-fault-management interfaces detail** command output displays continuity-check status as **enabled** and displays details of the MEP, it means that CFM was configured properly.
- If the **show oam ethernet connectivity-fault-management adjacencies** command output displays the state as **ok**, it indicates that the Continuity Check Protocol is up.

### Verifying MEP Continuity Using Ping

**Purpose** Verify access to MEPs under the same maintenance association.

**Action** From operational mode, enter the **ping ethernet** command.

```
user@host> ping ethernet maintenance-domain Customer-md maintenance-association
Customer-ma mep 201
PING to 2001:db8:2f:ca:8b:ab, Interface ge-0/0/1.0
64 bytes from 2001:db8:2f:ca:8b:ab: lbm_seq=0
64 bytes from 2001:db8:2f:ca:8b:ab: lbm_seq=1
64 bytes from 2001:db8:2f:ca:8b:ab: lbm_seq=2
64 bytes from 2001:db8:2f:ca:8b:ab: lbm_seq=3
--- ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
```

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269

## Creating a Maintenance Domain on a Security Device

**Supported Platforms** SRX Series

A maintenance domain consists of network entities such as operators, providers, and customers. A maintenance domain is a management space for managing and administering a network. A domain is owned and operated by a single entity and defined by the set of ports internal to it and at its boundary. Each maintenance domain is associated with a maintenance domain level from 0 through 7. Level allocation is based on the network hierarchy, where outer domains are assigned a higher level than the inner domains. You configure customer end points to have the highest maintenance domain level. The maintenance domain level is a mandatory parameter that indicates the nesting relationships between various maintenance domains.

To enable connectivity fault management (CFM) on an Ethernet interface, maintenance domains, maintenance associations, and maintenance association end points (MEPs) must be created and configured.

Starting in Junos OS Release 12.3X48-D65, on SRX210, SRX220, SRX240, and SRX550 devices, creating a maintenance domain for Ethernet OAM CFM is supported over VDSL and PPPoE interfaces in addition to Ethernet interfaces.

Starting in Junos OS Release 15.1X49-D80, creating a maintenance domain for Ethernet OAM CFM is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, creating a maintenance domain for Ethernet OAM CFM is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Creating a maintenance domain for Ethernet OAM CFM is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

To create a maintenance domain:

1. Specify a name for the maintenance domain.

   **[edit protocols oam ethernet connectivity-fault-management]**
   **user@host# set maintenance-domain *domain-name***

2. Specify a format for the maintenance domain name. If you do not specify a format, no name is configured.

   - A plain ASCII character string
   - A Domain Name System (DNS) format
   - A media access control (MAC) address plus a two-octet identifier in the range 0 through 65,535
   - None

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name*]**
   **user@host# set name-format *format***

For example, to specify the name format as a MAC address plus a two-octet identifier:

**[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name*]**

**user@host# set name-format *mac+2oct***

3. Configure the maintenance domain level, which is used to indicate the nesting relationship between this domain and other domains. Use a value from 0 through 7.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name*]**

   **user@host# set level *level-number***

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269
- Configuring the Continuity Check Protocol on a Security Device on page 295
- Configuring a Maintenance Domain MIP Half Function on a Security Device on page 294
- Creating a Maintenance Association on a Security Device on page 292
- Configuring a Maintenance Association End Point on a Security Device on page 293
- Configuring the Link Trace Protocol on a Security Device on page 297

## Creating a Maintenance Association on a Security Device

**Supported Platforms** SRX Series

In a connectivity fault management (CFM) maintenance domain, each service instance is called a maintenance association. A maintenance association can be thought of as a full mesh of maintenance association end points (MEPs) having similar characteristics.

Starting in Junos OS Release 12.3X48-D65, on SRX210, SRX220, SRX240, and SRX550 devices, creating a maintenance association for Ethernet OAM connectivity fault management is supported over VDSL and PPPoE interfaces in addition to Ethernet interfaces.

Starting in Junos OS Release 15.1X49-D80, creating a maintenance association for Ethernet OAM CFM is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, creating a maintenance association for Ethernet OAM CFM is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Creating a maintenance association for Ethernet OAM CFM is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

To create a maintenance association:

**[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name*]**
**user@host# set maintenance-association *ma-name***

NOTE: On SRX210, SRX220, SRX240, SRX300, SRX320, SRX340, SRX345, SRX550, SRX550M, and SRX650 devices, a maximum of seven maintenance associations are supported.

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269
- Creating a Maintenance Domain on a Security Device on page 291
- Configuring a Maintenance Domain MIP Half Function on a Security Device on page 294
- Configuring the Continuity Check Protocol on a Security Device on page 295
- Configuring a Maintenance Association End Point on a Security Device on page 293
- Configuring the Link Trace Protocol on a Security Device on page 297

## Configuring a Maintenance Association End Point on a Security Device

**Supported Platforms** SRX Series

Starting in Junos OS Release 12.3X48-D65, on SRX210, SRX220, SRX240, and SRX550 devices, configuring a maintenance association end point for Ethernet OAM CFM is supported over VDSL and PPPoE interfaces in addition to Ethernet interfaces.

Starting in Junos OS Release 15.1X49-D80, configuring a maintenance association end point for Ethernet OAM CFM is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, configuring a maintenance association end point for Ethernet OAM CFM is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Configuring a maintenance association end point for Ethernet OAM CFM is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

To configure a MEP:

1. Specify an ID for the MEP. The value can be from 1 through 8191.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name*]**
   **user@host# set mep *mep-id***

2. Enable MEP automatic discovery if you want to have the MEP accept continuity check messages (CCMs) from all remote MEPs of the same maintenance association.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name* mep *mep-id*]**
   **user@host# set auto-discovery**

3. Specify that CFM CCM packets be transmitted only in one direction for the MEP. That is, set the direction as down so that CCMs are transmitted only out of (not into) the interface configured on this MEP.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name* mep *mep-id*]**
   **user@host# set direction down**

4. Specify the logical interface to which the MEP is attached. It can be either an access interface or a trunk interface. If you specify a trunk interface, the VLAN associated with that interface must have a VLAN ID.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name* mep *mep-id*]**
   **user@host# set interface *interface-name***

5. Configure a remote MEP from which CCMs are expected. If automatic discovery is not enabled, the remote MEP must be configured under the **mep** statement; otherwise, the CCMs from the remote MEP will be treated as errors.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name maintenance-association ma-name mep mep-id]**
   **user@host# set remote-mep mep-id**

NOTE: You cannot configure MEPs at different levels for the same VLANs.

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269
- Creating a Maintenance Domain on a Security Device on page 291
- Configuring a Maintenance Domain MIP Half Function on a Security Device on page 294
- Creating a Maintenance Association on a Security Device on page 292
- Configuring the Continuity Check Protocol on a Security Device on page 295
- Configuring the Link Trace Protocol on a Security Device on page 297

## Configuring a Maintenance Domain MIP Half Function on a Security Device

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D80, configuring a maintenance domain MIP half function for Ethernet OAM connectivity fault management is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, configuring a maintenance domain MIP half function for Ethernet OAM connectivity fault management is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Configuring a maintenance domain MIP half function for Ethernet OAM connectivity fault management is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

MIP half function (MHF) divides the maintenance association intermediate point (MIP) functionality into two unidirectional segments, improves visibility with minimal configuration, and improves network coverage by increasing the number of points that can be monitored. MHF extends monitoring capability by responding to loopback and Link Trace messages to help isolate faults. Whenever a MIP is configured, the MIP half function value for all maintenance domains and maintenance associations must be the same.

To configure the MIP half function:

**[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name*]**
**user@host# set mip-half-function *default***

NOTE:

- If SRX340, or SRX345 devices are configured as MIPs, ensure that a static MAC is configured in the Ethernet switching table with the next-hop interface to the MEP MAC.
- You cannot configure MIP in a nondefault domain.
- In Q-in-Q mode, double tag packets are not retained by MIP.
- A maximum of 116 MIPs can be configured on a device.

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269
- Creating a Maintenance Domain on a Security Device on page 291
- Creating a Maintenance Association on a Security Device on page 292
- Configuring the Continuity Check Protocol on a Security Device on page 295
- Configuring a Maintenance Association End Point on a Security Device on page 293
- Configuring the Link Trace Protocol on a Security Device on page 297

## Configuring the Continuity Check Protocol on a Security Device

**Supported Platforms** SRX Series

The Continuity Check Protocol is used for fault detection by a maintenance association end point (MEP) within a maintenance association. The MEP periodically sends continuity check multicast messages. The receiving MEPs use the continuity check messages (CCMs) to build a MEP database of all MEPs in the maintenance association.

Starting in Junos OS Release 12.3X48-D65, on SRX210, SRX220, SRX240, and SRX550 devices, the continuity check protocol for Ethernet Operation, Administration, and Management (OAM) connectivity fault management is supported over VDSL and PPPoE interfaces in addition to Ethernet interfaces.

Starting in Junos OS Release 15.1X49-D80, the continuity check protocol for Ethernet OAM CFM is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, the continuity check protocol for Ethernet OAM CFMis supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

The continuity check protocol for Ethernet OAM CFM is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

To configure the Continuity Check Protocol:

1. Enable the Continuity Check Protocol.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name*]**
   **user@host# set continuity-check**

2. Specify the continuity check hold interval. The hold interval is the number of minutes to wait before flushing the MEP database if no updates occur. The default value is 10 minutes (not supported in Junos OS Release 12.3X48-D60).

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name* continuity-check]**
   **user@host# set hold-interval *number***

3. Specify the CCM interval. The interval is the time between the transmission of CCMs. You can specify 10 minutes (10m), 1 minute (1m), 10 seconds (10s), 1 second (1s), or 100 milliseconds (100ms).

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name* continuity-check]**
   **user@host# set interval *number***

4. Specify the number of CCMs (that is, protocol data units) that can be lost before the MEP is marked as down. The default number of protocol data units (PDUs) is 3.

   **[edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name* continuity-check]**
   **user@host# set loss-threshold *number***

NOTE: If the CCM interval is 100 milliseconds, only four MEPs are supported on a device.

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269
- Creating a Maintenance Domain on a Security Device on page 291
- Creating a Maintenance Association on a Security Device on page 292
- Configuring a Maintenance Domain MIP Half Function on a Security Device on page 294
- Configuring the Link Trace Protocol on a Security Device on page 297

## Configuring the Link Trace Protocol on a Security Device

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D80, configuring the Link Trace protocol for Ethernet OAM connectivity fault management is supported on SRX1500 devices.

Starting in Junos OS Release 15.1X49-D75, configuring the Link Trace protocol for Ethernet OAM connectivity fault management is supported on SRX300, SRX320, SRX340, SRX345, and SRX550M devices.

Configuring the Link Trace protocol for Ethernet OAM connectivity fault management is not supported from Junos OS Release 15.1X49-D40 to Junos OS Release 15.1X49-D70.

The Link Trace protocol is used for path discovery between a pair of maintenance points. Link Trace Messages (LTMs) are triggered by an administrator using the **traceroute ethernet** command to verify the path between a pair of MEPs under the same maintenance association. LTMs can also be used to verify the path between a MEP and a MIP under the same maintenance domain.

To configure the Link Trace protocol:

1. Configure the Link Trace path age timer. If no response to a Link Trace request is received, the request and response entries are deleted after the age timer expires.

   **[edit protocols oam ethernet connectivity-fault-management]**
   **user@host# set linktrace age *time***

2. Configure the number of Link Trace Reply (LTR) entries to be stored per Link Trace request.

   **[edit protocols oam ethernet connectivity-fault-management]**
   **user@host# set linktrace path-database-size *path-database-size***

**Related Documentation**

- Understanding Ethernet OAM Connectivity Fault Management on page 269
- Creating a Maintenance Domain on a Security Device on page 291

CHAPTER 23

# Configuring Ethernet OAM Link Fault Management



## Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D70, Ethernet OAM link fault management for SRX Series services gateways is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

The Ethernet interfaces on SRX Series devices support the IEEE 802.3ah standard for Operation, Administration, and Maintenance (OAM). The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters. The IEEE 802.3ah standard meets the requirement for OAM capabilities as Ethernet moves from being solely an enterprise technology to a WAN and access technology, and the standard remains backward-compatible with existing Ethernet technology.

NOTE: **For SRX550M devices, LFM is supported only on devices that have 16-port or 24-port GPIMs.**

The following OAM LFM features are supported:

- Discovery and link monitoring—The discovery process is triggered automatically when OAM is enabled on the interface. The discovery process permits Ethernet interfaces to discover and monitor the peer on the link if it also supports the IEEE 802.3ah standard. In active mode, the interface discovers and monitors the peer on the link if the peer also supports IEEE 802.3ah OAM functionality. In passive mode, the peer initiates the discovery process. After the discovery process has been initiated, both sides participate

in discovery. The device performs link monitoring by sending periodic OAM protocol data units (PDUs) to advertise OAM mode, configuration, and capabilities.

You can specify the number of OAM PDUs that an interface can miss before the link between peers is considered down.

- Remote fault detection—Remote fault detection uses flags and events. Flags convey Link Fault (a loss of signal), Dying Gasp (an unrecoverable condition such as a power failure), and Critical Event (an unspecified vendor-specific critical event). You can specify the periodic OAM PDU sending interval for fault detection. SRX Series devices use the Event Notification OAM PDU to notify the remote OAM device when a problem is detected. You can specify the action to be taken by the system when the configured link-fault event occurs.
- Remote loopback—Remote loopback mode ensures link quality between the device and a remote peer during installation or troubleshooting. In this mode, when the interface receives a frame that is not an OAM PDU or a pause frame, it sends it back on the same interface on which it was received. The link appears to be in the active state. You can use the returned loopback acknowledgement to test delay, jitter, and throughput.

  Junos OS can place a remote data terminal equipment (DTE) into loopback mode (if remote loopback mode is supported by the remote DTE). When you place a remote DTE into loopback mode, the interface receives the remote loopback request and puts the interface into remote loopback mode. When the interface is in remote loopback mode, all frames except OAM PDUs are looped back without any changes made to the frames. OAM PDUs continue to be sent and processed.

Table 37 on page 300 lists the interfaces modes supported.

### Table 37: Supported Interface Modes

| Interfaces | Mode |
|---|---|
| Physical interface (fe/ge) | Family<br>• ccc<br>• ethernet-switching<br>• inet6<br>• inet<br>• iso<br>• mpls<br>• tcc |
| | IFD encapsulations<br>• ethernet-ccc<br>• extended-vlan-ccc (IFD vlan-tagging mode)<br>• ethernet-tcc<br>• extended-vlan-tcc |

Table 37: Supported Interface Modes *(continued)*

| Interfaces | Mode |
|---|---|
| Aggregated Ethernet interface (Static or LACP lag) | Family<br>• ethernet-switching<br>• inet<br>• mpls<br>• iso<br>• inet6 |
| | IFD encapsulations<br>• ethernet-ccc<br>• extended-vlan-ccc (IFD vlan-tagging mode)<br>• vlan-ccc |

**Related Documentation**

- Example: Configuring Ethernet OAM Link Fault Management on page 301

## Example: Configuring Ethernet OAM Link Fault Management

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D70, configuring Ethernet OAM link fault management is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

The Ethernet interfaces on the SRX Series devices support the IEEE 802.3ah standard for Operation, Administration, and Maintenance (OAM). The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters.

This example describes how to enable and configure OAM LFM on a Gigabit Ethernet or Fast Ethernet interface:



### Requirements

This example uses the following hardware and software components:

- Junos OS Release 12.1 R2 or later for SRX Series Services Gateways
- Any two models of SRX Series devices connected directly

Before you begin:

- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces as necessary. See *Example: Creating an Ethernet Interface*.
- Ensure that you configure the interfaces as per the interface modules listed in "Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways" on page 299

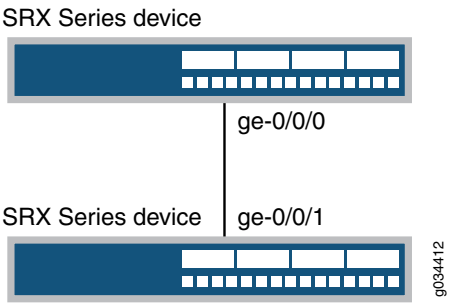
## Overview

The Ethernet interfaces on the SRX Series devices support the IEEE 802.3ah standard for Operation, Administration, and Maintenance (OAM). The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters.

This example uses two SRX Series devices connected directly. Before you begin configuring Ethernet OAM LFM on these two devices, connect the two devices directly through supported interfaces. See "Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways" on page 299.

Figure 13 on page 302 shows the topology used in this example.

### Figure 13: Ethernet LFM with SRX Series Devices




NOTE: For more information about configuring Ethernet OAM Link Fault Management, see Junos® OS Ethernet Interfaces.

## Configuration

To configure Ethernet OAM LFM, perform these tasks:

- Configuring Ethernet OAM Link Fault Management on Device 1 on page 302
- Configuring Ethernet OAM Link Fault Management on Device 2 on page 303

### Configuring Ethernet OAM Link Fault Management on Device 1

**CLI Quick Configuration**

To quickly configure this example, copy the following command, paste it into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the command into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set protocols oam ethernet link-fault-management interface ge-0/0/0
set protocols oam ethernet link-fault-management interface ge-0/0/0 pdu-interval 800
set protocols oam ethernet link-fault-management interface ge-0/0/0 link-discovery
  active
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the Junos OS CLI User Guide.

To configure Ethernet OAM LFM on device 1:

1. Enable IEEE 802.3ah OAM support.

   ```
   [edit protocols oam ethernet link-fault-management]
   user@device1# set interface ge-0/0/0
   ```

2. Set the periodic OAM PDU-sending interval (in milliseconds) for fault detection.

   ```
   [edit protocols oam ethernet link-fault-management]
   user@device1# set interface pdu-interval 800
   ```

3. Specify that the interface initiates the discovery process.

   ```
   [edit protocols oam ethernet link-fault-management]
   user@device1# set interface ge-0/0/0 link-discovery active
   ```

**Results**

From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@device1# show protocols
protocols {
  oam {
    ethernet {
      link-fault-management {
        interface ge-0/0/0 {
          pdu-interval 800;
          link-discovery active;
        }
      }
    }
  }
}
```

## Configuring Ethernet OAM Link Fault Management on Device 2

**CLI Quick Configuration**

To quickly configure this example, copy the following command, paste it into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the command into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set protocols oam ethernet link-fault-management interface ge-0/0/1
set protocols oam ethernet link-fault-management interface ge-0/0/1 pdu-interval 800
set protocols oam ethernet link-fault-management interface ge-0/0/1 negotiation-options
  allow-remote-loopback
```

**Step-by-Step Procedure**

To configure Ethernet OAM LFM on device 2:

1. Enable OAM on the peer interface.

   ```
   [edit protocols oam ethernet link-fault-management]
   user@device2# set interface ge-0/0/1
   ```

2. Set the periodic OAM PDU-sending interval (in milliseconds) for fault detection.

   ```
   [edit protocols oam ethernet link-fault-management]
   user@device2# set interface ge-0/0/1 pdu-interval 800
   ```

3. Enable remote loopback support for the local interface.

   ```
   [edit protocols oam ethernet link-fault-management]
   user@device2# set interface ge-0/0/1 negotiation-options allow-remote-loopback
   ```

**Results**

From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@device2# show protocols
protocols {
  oam {
    ethernet {
      link-fault-management {
        interface ge-0/0/1 {
          negotiation-options {
            allow-remote-loopback;
          }
        }
      }
    }
  }
}
```

## Verification

### Verify the OAM LFM Configuration

**Purpose**

Verify that OAM LFM is configured properly.

**Action**

From operational mode, enter the **show oam ethernet link-fault-management** command.

```
user@device1> show oam ethernet link-fault-management

Interface: ge-0/0/0.0
Status: Running, Discovery state: Send Any
Peer address: 2001:bd8:00:31
Flags:Remote-Stable Remote-State-Valid Local-Stable 0x50
Remote entity information:
Remote MUX action: forwarding, Remote parser action: forwarding
Discovery mode: active, Unidirectional mode: unsupported
Remote loopback mode: supported, Link events: supported
Variable requests: unsupported
```

**Meaning** The output displays the MAC address and the discovery state is **Send Any** if OAM LFM has been configured properly.

**Related Documentation**

- Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways on page 299

## Example: Configuring Remote Loopback Mode on VDSL Interfaces on a Security Device

**Supported Platforms** SRX Series

Starting in Junos OS Release 15.1X49-D110, configuring remote loopback mode in Ethernet OAM link fault management (LFM) on a VDSL interface is supported on SRX320, SRX340, SRX345, and SRX550M devices.

This example describes the following configuration scenarios:

Starting in Junos OS Release 12.3X48-D65, configuring remote loopback mode in Ethernet OAM link fault management (LFM) on a VDSL interface is supported on SRX210, SRX220, SRX240, and SRX550 devices.

This example describes the following configuration scenarios:

- Scenario 1: Configuring remote loopback mode on a VDSL interface.
- Scenario 2: Configuring remote loopback mode on a VDSL interface acting as a PPPOE's underlying interface.



### Requirements

This example uses the following hardware and software components:

- Junos OS Release 15.1X49-D110 or later for SRX Series Services Gateways

- An SRX 210/220/240/320/340/345/550/550M device connected with a DSLAM

Before you begin:

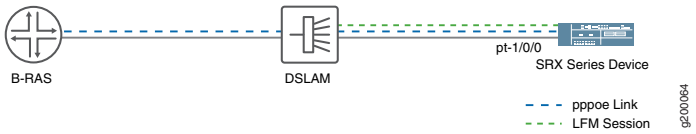
- Establish basic connectivity. See the Getting Started Guide for your device.
- Configure network interfaces as necessary. See *Example: Configuring VDSL2 Interfaces (Basic)*.
- Ensure that you configure the interfaces as per the interface modules listed in "Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways" on page 299
- Ensure that you configure PPPOE as per the instructions listed in *Example: Configuring PPPoE Interfaces*

## Overview

This example uses an SRX Series device connected to a DSLAM. Before you begin configuring Ethernet OAM LFM on these two devices, connect the two devices directly through supported interfaces. See "Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways" on page 299.

Figure 13 on page 302 shows the topology used in this example.

### Figure 14: Ethernet LFM with SRX Series Devices




NOTE: For more information about configuring Ethernet OAM Link Fault Management, see Junos® OS Ethernet Interfaces.

## Configuration for Scenario 1

To configure remote loopback mode on a VDSL interface, perform these tasks:

- Configuring Remote Loopback Mode on a VDSL interface of an SRX Series Device on page 306

### Configuring Remote Loopback Mode on a VDSL interface of an SRX Series Device

**CLI Quick Configuration**

To quickly configure this example, copy the following command, paste it into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the command into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set protocols oam ethernet link-fault-management interface pt-1/0/0
```

```
set protocols oam ethernet link-fault-management interface pt-1/0/0 negotiation-options
  allow-remote-loopback
```

**Step-by-Step Procedure**

To configure remote loopback mode on a VDSL interface of an SRX Series device:

1. Enable OAM on a VDSL interface.

```
[edit protocols oam ethernet link-fault-management]
user@device2# set interface pt-1/0/0
```

2. Enable remote loopback support for the interface.

```
[edit protocols oam ethernet link-fault-management]
user@device2# set interface pt-1/0/0 negotiation-options allow-remote-loopback
```

**Results**

From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@device2# show protocols
protocols {
  oam {
    ethernet {
      link-fault-management {
        interface pt-1/0/0 {
          negotiation-options {
            allow-remote-loopback;
          }
        }
      }
    }
  }
}
```

## Configuration for Scenario 2

To configure remote loopback mode on a PPPOE's underlying interface, perform these tasks:

- Configuring Remote Loopback Mode on a PPPOE's underlying interface on page 307

### Configuring Remote Loopback Mode on a PPPOE's underlying interface

**CLI Quick Configuration**

To quickly configure this example, copy the following command, paste it into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the command into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces pp0 unit 0 pppoe-options underlying-interface pt-1/0/0
set protocols oam ethernet link-fault-management interface pt-1/0/0 link-discovery
  active
```

```
set protocols oam ethernet link-fault-management interface pt-1/0/0 negotiation-options
  allow-remote-loopback
```

**Step-by-Step Procedure**

To configure remote loopback mode on a PPPOE's underlying interface:

1. Create the PPPoE interface pp0 and specify the logical PT interface pt-1/0/0 as the underlying interface.

   ```
   [edit protocols oam ethernet link-fault-management]
   user@device2# set interfaces pp0 unit 0 pppoe-options underlying-interface
     pt-1/0/0
   ```

2. Specify that the interface initiates the discovery process.

   ```
   user@device2# set protocols oam ethernet link-fault-management interface
     pt-1/0/0 link-discovery active
   ```

3. Enable remote loopback mode.

   ```
   user@device2# set protocols oam ethernet link-fault-management interface
     pt-1/0/0 negotiation-options allow-remote-loopback
   ```

**Results**

From configuration mode, confirm your configuration by entering the **show protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@device2# show protocols
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface pt-1/0/0 {
                    link-discovery active;
                    negotiation-options {
                        allow-remote-loopback;
                    }
                }
            }
        }
    }
}
```

## Verification

### Verify the OAM LFM Configuration

**Purpose**

Verify that OAM LFM is configured properly.

**Action**

From operational mode, enter the **show oam ethernet link-fault-management** command.

user@device1> **show oam ethernet link-fault-management**

```
Interface: pt-1/0/0.0
Status: Running, Discovery state: Send Any
Transmit interval: 300ms, PDU threshold: 3 frames, Hold time: 900ms
Peer address: 2001:db8:e5:b9:c8:ed
Flags:Remote-Stable Remote-State-Valid Local-Stable 0x50
Loopback tracking: Disabled, Loop status: Unknown
Remote entity information:
Remote MUX action: forwarding, Remote parser action: forwarding
Discovery mode: active, Unidirectional mode: unsupported
Remote loopback mode: unsupported, Link events: supported
Variable requests: unsupported
```

**Meaning** The output displays the MAC address and the discovery state is **Send Any** if OAM LFM has been configured properly.

**Related Documentation**

- Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways on page 299

PART 4

# Configuration Statements and Operational Commands

CHAPTER 24

# Configuration Statements

## bpdu-block

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**

```
bpdu-block {
  interface (interface-name disable | all);
  disable-timeout seconds;
}
```

**Hierarchy Level** [edit protocols layer2-control ]

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70.

**Description** Enable BPDU blocking on an interface.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Understanding Root Protection for STP, RSTP, and MSTP on page 171

## bpdu-destination-mac-address

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** bpdu-destination-mac-address provider-bridge-group;

**Hierarchy Level** [edit protocols mvrp],
[edit routing-instances *routing-instance-name* protocols mvrp] (for virtual switch instance type)

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** For Multiple VLAN Registration Protocol (MVRP) configurations, specify the multicast address for MVRP. If configured, Junos OS uses the provider MVRP multicast MAC address; otherwise, it uses the customer MVRP multicast MAC address.

**Default** By default, the customer MVRP MAC address is used.

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

**Related Documentation**

- Configuring Multiple VLAN Registration Protocol (MVRP) to Manage Dynamic VLAN Registration on Security Devices on page 128

## code-points (CoS)

**Supported Platforms** NFX Series, SRX Series, vSRX

**Syntax** code-points [ *aliases* ] [ *bit-patterns* ];

**Hierarchy Level** [edit class-of-service classifiers (dscp) *classifier-name* forwarding-class *class-name* loss-priority *level* ]

**Release Information** Statement introduced in Junos OS Release 12.1X44 for the SRX Series.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

**Description** Configure one or more code-point aliases or bit sets to apply to a forwarding class.

NOTE: OCX Series switches do not support MPLS, and therefore, do not support EXP code points or code point aliases.

**Options** ***aliases***—Name of the alias or aliases.

***bit-patterns***—Value of the code-point bits, in decimal form.

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

**Related Documentation**

- *Understanding Interfaces*
- Example: Configuring BA Classifiers on Transparent Mode Security Devices on page 59

## destination-address (Security Policies)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
destination-address {
  [address];
  any;
  any-ipv4;
  any-ipv6;
}
```

**Hierarchy Level** [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* match]

[edit security policies global policy *policy-name* match]

**Release Information** Statement introduced in Junos OS Release 8.5. Support for IPv6 addresses added in Junos OS Release 10.2. Support for IPv6 addresses in active/active chassis cluster configurations (in addition to the existing support of active/passive chassis cluster configurations) added in Junos OS Release 10.4. Support for wildcard addresses added in Junos OS Release 11.1.

**Description** Define the matching criteria. You can specify one or more IP addresses, address sets, or wildcard addresses. You can specify wildcards **any**, **any-ipv4**, or **any-ipv6**.

**Options** ***address***—IP address (**any**, **any-ipv4**, **any-ipv6**), IP address set, or address book entry, or wildcard address (represented as A.B.C.D/wildcard-mask). You can configure multiple addresses or address prefixes separated by spaces and enclosed in square brackets.

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

**Related Documentation**

- *Security Policies Overview*

## disable-timeout (Spanning Trees)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** disable-timeout *seconds*;

**Hierarchy Level** [edit protocols layer2-control bpdu-block]

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70.

**Description** For interfaces configured for BPDU protection, specify the amount of time an interface is disabled by BPDU blocking. If this option is not configured, the interface is not periodically checked and remains disabled.

**Default** The disable timeout is not enabled.

**Options** *seconds*—Amount of time, in seconds, the interface receiving BPDUs protect is disabled. The range is 10 through 3600 seconds.

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

**Related Documentation**

- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Understanding Root Protection for STP, RSTP, and MSTP on page 171
- Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees on page 172

## domain-type (Bridge Domains)

**Supported Platforms** SRX Series

**Syntax** domain-type bridge;

**Hierarchy Level** [edit bridge-domains *bridge-domain-name*]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Define the domain type **bridge** for a Layer 2 bridge domain. There is only one domain type **bridge**, that can be configured on SRX Series devices. Domain type **bridge** is not enabled by default. An SRX Series device operates in the Layer 2 transparent mode when all physical bridge domains on the device are partitioned into logical bridge domains.

Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the CLI **domain-type** is not available.

NOTE: Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the hierarchy **[edit bridge-domains bridge-domain-name]** is renamed to **[edit vlans vlan-name]**. For detailed information about the modified hierarchies, see "Enhanced Layer 2 CLI Configuration Statement and Command Changes for Security Devices" on page 34.

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

**Related Documentation**

- Understanding Layer 2 Transparent Mode on SRX Devices on page 25

## dot1x

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**

```
dot1x {
  authenticator {
    authentication-profile-name access-profile-name;
    interface (all | <interface-names>) {
      disable;
      guest-vlan (vlan-tag | vlan-name);
      lldp-med-bypass;
      mac-radius <restrict | flap-on-disconnect>;
      maximum-requests number;
      no-reauthentication;
      quiet-period seconds;
      reauthentication interval;
      retries retries-number;
      server-fail (deny | permit | use-cache | vlan-name vlan-name );
      server-reject-vlan vlan-name;
      server-timeout seconds;
      supplicant (single | single-secure | multiple);
      supplicant-timeout seconds;
      transmit-period seconds
    }
    no-mac-table-binding;
    use-vlan-name <use-vlan-id | use-vlan-name>;
    static  mac-address ;
  traceoptions {
      file filename  <files number> <no-world-readable | world-readable> <size size>;
      flag;
  }
}
```

**Hierarchy Level** [edit protocols]

**Release Information** Command introduced in Junos OS Release 15.1X49-D80.

**Description** Configure 802.1X authentication for port-based network access control (PNAC). 802.1X authentication is supported on interfaces that are members of private VLANs (PVLANs).

Define tracing operations for the 802.1X protocol authentication.

**Default** 802.1X authentication is disabled.

**Options** **authentication-profile-name** *access-profile-name*—Name of the access profile to use for authentication.

**all**—Configure all interfaces for 802.1X authentication.

*interface-names*—List of names of interfaces to configure for 802.1X authentication.

**guest-vlan** *vlan-tag* —VLAN tag identifier of the guest VLAN.

**guest-vlan** *vlan-name*—Name of the guest VLAN.

**mac-radius flap-on-disconnect**—(Optional) Reset an interface on receiving a disconnect request.

**mac-radius restrict**—(Optional) Bypass dot1x authentication. This restricts authentication to MAC RADIUS.

**maximum-requests** *number*—Maximum number of times an EAPOL request packet is retransmitted to the supplicant before the authentication session times out.

**quiet-period** *seconds*—Number of seconds the interface remains in the wait state.

**reauthentication** *interval*—Sets the periodic reauthentication time interval in seconds.

**retries** *number*—Number of retries after which port is placed into wait state.

**deny**—Force fail the supplicant authentication. No traffic flows through the interface.

**permit**—Force succeed the supplicant authentication. Traffic flows through the interface as if it were successfully authenticated by the RADIUS server.

**use-cache**—Force succeed the supplicant authentication only if it was previously authenticated successfully. This action ensures that already authenticated supplicants are not affected.

**server-fail** *vlan-tag* —Move supplicant on the interface to the VLAN specified by this numeric identifier.

**server-fail** *vlan-name*—Move supplicant on the interface to the VLAN specified by this name.

**server-fail** *seconds*—The time interval, in seconds, during which the device does not attempt to contact the authentication server to reauthenticate a client that has already been authenticated using server fail fallback.

**server-timeout** *seconds*—Amount of time a port waits for a reply when relaying a response from the supplicant to the authentication server before timing out and invoking the server-fail action.

**single**—Authenticates only the first client that connects to an authenticator port.

**single-secure**—Authenticates only one client to connect to an authenticator port.

**multiple**—Authenticates multiple clients individually on one authenticator port.

**supplicant-timeout** *seconds*—Number of seconds the port waits for a response when relaying a request from the authentication server to the supplicant before resending the request.

**transmit-period** *seconds*—Number of seconds the port waits before retransmitting the initial EAPOL PDUs to the supplicant.

*mac-address*—The MAC address of the device for which 802.1X authentication must be bypassed and the device permitted access to the port.

**file** *filename*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory **/var/log**.

**files** *number*—(Optional) Maximum number of trace files.

**flag** *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements. You can include the following flags:

- **all**—All tracing operations.
- **config-internal**—Trace internal configuration operations.
- **eapol**—Trace EAPOL packets transmitted and received.
- **general**—Trace general operations.
- **normal**—Trace normal operations.
- **parse**—Trace reading of the configuration.
- **state**—Trace protocol state changes.
- **task**—Trace protocol task operations.
- **timer**—Trace protocol timer operations.
- **vlan**—Trace VLAN transactions.

**no-world-readable**—(Optional) Restrict file access to the user who created the file.

**size** *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB).

**world-readable**—(Optional) Enable unrestricted file access.

**Required Privilege Level**

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

**Related Documentation**

- Understanding 802.1X Port-Based Network Authentication on page 249
- clear dot1x on page 391

## encapsulation (Interfaces)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
encapsulation (ether-vpls-ppp | ethernet-bridge | ethernet-ccc | ethernet-tcc | ethernet-vpls
  | extended-frame-relay-ccc | extended-frame-relay-tcc | extended-vlan-bridge |
  extended-vlan-ccc | extended-vlan-tcc | extended-vlan-vpls | frame-relay-port-ccc |
  vlan-ccc | vlan-vpls);
```

**Hierarchy Level** [edit interfaces *interface-name* unit *logical-unit-number* ]

**Release Information** Statement introduced in Junos OS Release 9.5.

**Description** Specify logical link layer encapsulation.

**Options**

- **frame-relay**—Configure a Frame Relay encapsulation when the physical interface has multiple logical units, and the units are either point to point or multipoint.
- **multilink-frame-relay-uni-nni**—Link services interfaces functioning as FRF.16 bundles can use Multilink Frame Relay UNI NNI encapsulation.
- **ppp**—For normal mode (when the device is using only one ISDN B-channel per call). Point-to-Point Protocol is for communication between two computers using a serial interface.
- **ppp-over-ether**—This encapsulation is used for underlying interfaces of pp0 interfaces.

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

**Related Documentation**

- *Understanding Physical Encapsulation on an Interface*

## ethernet (Chassis Cluster)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
ethernet {
  device-count number;
  lacp {
    link-protection {
      non-revertive;
    }
    system-priority number;
  }
}
```

**Hierarchy Level** [edit chassis aggregated-devices]

**Release Information** Statement introduced in Junos OS Release 10.2.

**Description** Configure properties for aggregated Ethernet devices.

**Options** The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- *cluster (Chassis)*
- Example: Configuring Aggregated Ethernet Device with LAG and LACP (CLI Procedure) on page 186

## ethernet-switching

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
ethernet-switching {
  block-non-ip-all;
  bpdu-vlan-flooding;
  bypass-non-ip-unicast;
  no-packet-flooding {
    no-trace-route;
  }
}
```

**Hierarchy Level** [edit security flow]

**Release Information** Statement introduced in Junos OS Release 9.5.

**Description** Changes default Layer 2 forwarding behavior for Ethernet switching flow configuration.

**Options**

- **block-non-ip-all**—Block all Layer 2 non-IP and non-ARP traffic, including multicast and broadcast traffic. This option is not enabled by default.
- **bpdu-vlan-flooding**—Set 802.1D bridge protocol data unit (BPDU) flooding based on VLAN. This option is not enabled by default.
- **bypass-non-ip-unicast**—Allow all Layer 2 non-IP traffic to pass through the device. This option is not enabled by default.
- **no-packet-flooding**—Stop IP flooding and send ARP or ICMP requests to discover the destination MAC address for a unicast packet. This option is not enabled by default.

NOTE: On all SRX Series devices in transparent mode, packet flooding is enabled by default. If you have manually disabled packet flooding with the set security flow ethernet-switching no-packet-flooding command, then multicast packets such as OSPFv3 hello packets are dropped.

- **no-trace-route**—Do not send ICMP requests to discover the destination MAC address for a unicast packet. Only ARP requests are sent. This option only allows the device to discover the destination MAC address for a unicast packet if the destination IP address is in the same subnetwork as the ingress IP address. This option is not enabled by default.

NOTE: The block-non-ip-all and bypass-non-ip-unicast options cannot be configured at the same time.

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

**Related Documentation**
- *Juniper Networks Devices Processing Overview*

## family inet (Interfaces)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
inet {
  accounting {
    destination-class-usage;
    source-class-usage {
      input;
      output;
    }
  }
  address (source–address/prefix) {
    arp destination-address {
      (mac mac-address | multicast-mac multicast-mac-address);
      publish publish-address;
    }
    broadcast address;
    preferred;
    primary;
    vrrp-group group-id {
      (accept-data | no-accept-data);
      advertise-interval seconds;
      advertisements-threshold number;
      authentication-key key-value;
      authentication-type (md5 | simple);
      fast-interval milliseconds;
      inet6-advertise-interval milliseconds
      (preempt <hold-time seconds> | no-preempt );
      priority value;
      track {
        interface interface-name {
          bandwidth-threshold bandwidth;
          priority-cost value;
        }
        priority-hold-time seconds;
        route route-address{
          routing-instance routing-instance;
          priority-cost value;
        }
      }
      virtual-address [address];
      virtual-link-local-address address;
      vrrp-inherit-from {
        active-group value;
        active-interface interface-name;
      }
    }
    web-authentication {
      http;
      https;
      redirect-to-https;
    }
  }
```

```
dhcp {
  client-identifier {
    (ascii string | hexadecimal string);
  }
  lease-time (length | infinite);
  retransmission-attempt value;
  retransmission-interval seconds;
  server-address server-address;
  update-server;
  vendor-id vendor-id ;
}
dhcp-client {
  client-identifier {
    prefix {
      host-name;
      logical-system-name;
      routing-instance-name;
    }
    use-interface-description (device | logical);
    user-id (ascii string| hexadecimal string);
  }
  lease-time (length | infinite);
  retransmission-attempt value;
  retransmission-interval seconds;
  server-address server-address;
  update-server;
  vendor-id vendor-id ;
}
filter {
  group number;
  input filter-name;
  input-list [filter-name];
  output filter-name;
  output-list [filter-name];
}
mtu value;
no-neighbor-learn;
no-redirects;
policer {
  arp arp-name;
  input input-name;
  output output-name;
}
primary;
rpf-check {
  fail-filter filter-name;
  mode {
    loose;
  }
}
sampling {
  input;
  output;
  simple-filter;
}
targeted-broadcast {
```

```
    (forward-and-send-to-re |forward-only);
  }
  unnumbered-address {
    interface-name;
    preferred-source-address preferred-source-address;
  }
}
```

**Hierarchy Level** [edit interfaces *interface* unit *unit* ]

**Release Information** Statement supported in Junos 10.2 for SRX Series devices.

**Description** Assign an IP address to a logical interface.

**Options** *ipaddress*—Specify the IP address for the interface. The remaining statements are explained separately.

NOTE: You use family inet to assign an IPv4 address. You use family inet6 to assign an IPv6 address. An interface can be configured with both an IPv4 and IPv6 address.

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

**Related Documentation**

- *Understanding Interfaces*

## family inet6

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
inet6 {
  accounting {
    destination-class-usage;
    source-class-usage {
      input;
      ouput;
    }
  }
  address source–address/prefix {
    eui-64;
    ndp address {
      (mac mac-address | multicast-mac multicast-mac-address);
      publish;
    }
    preferred;
    primary;
    vrrp-inet6-group group_id {
      (accept-data | no-accept-data);
      advertisements-threshold number;
      authentication-key value;
      authentication-type (md5 | simple);
      fast-interval milliseconds;
      inet6-advertise-interval milliseconds;
      (preempt <hold-time seconds>| no-preempt );
      priority value;
      track {
        interface interface-name {
          bandwidth-threshold value;
          priority-cost value;
        }
        priority-hold-time seconds;
        route route-address{
          routing-instance routing-instance;
        }
      }
      virtual-inet6-address [address];
      virtual-link-local-address address;
      vrrp-inherit-from {
        active-group value;
        active-interface interface-name;
      }
    }
    web-authentication {
      http;
      https;
      redirect-to-https;
    }
  }
  (dad-disable | no-dad-disable);
  dhcpv6-client {
```

```
    client-ia-type (ia-na | ia-pd);
    client-identifier duid-type (duid-ll | duid-llt | vendor);
    client-type (autoconfig | stateful);
    rapid-commit;
    req-option (dns-server | domain | fqdn | nis-domain | nis-server | ntp-server | sip-domain
      | sip-server |time-zone | vendor-spec);
    retransmission-attempt number;
    update-router-advertisement {
      interface interface-name;
    }
    update-server;
  }
  filter {
    group number;
    input filter-name;
    input-list [filter-name];
    output filter-name;
    output-list [filter-name];
  }
  mtu value;
  nd6-stale-time seconds;
  no-neighbor-learn;
  policer {
    input input-name;
    output output-name;
  }
  rpf-check {
    fail-filter filter-name;
    mode {
      loose;
    }
  }
  sampling {
    input;
    output;
  }
  unnumbered-address {
    interface-name;
    preferred-source-address preferred-source-address;
  }
}
(proxy-ndp | proxy-dad) {
  (apply-groups | apply-groups-except | interface-restricted)
  (no-proxy-on-resolve | no-proxy-on-resolve)
}
```

**Hierarchy Level** [edit interfaces *interface* unit *unit* ]

**Release Information** Statement supported in Junos 10.2 for SRX Series devices.

**Description** Assign an IPV6 address to a logical interface.

**Options** *ipaddress*—Specify the IP address for the interface. The remaining statements are explained separately.

NOTE: You use family **inet6** to assign an IPv6 address. You use family **inet** to assign an IPv4 address. An interface can be configured with both an IPv4 and IPv6 address.

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

**Related Documentation**

- *Understanding Interfaces*

## flow (Security Flow)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
flow {
  aging {
    early-ageout seconds;
    high-watermark percent;
    low-watermark percent;
  }
  allow-dns-reply;
  ethernet-switching {
    block-non-ip-all;
    bpdu-vlan-flooding;
    bypass-non-ip-unicast;
    no-packet-flooding {
      no-trace-route;
    }
  }
  force-ip-reassembly;
  ipsec-performance-acceleration;
  load distribution {
    session-affinity ipsec;
  }
  packet-log {
    enable;
    throttle-interval;
    packet-filter <filter-name>;
  }
  pending-sess-queue-length (high | moderate | normal);
  route-change-timeout seconds;
  syn-flood-protection-mode (syn-cookie | syn-proxy);
  tcp-mss {
    all-tcp mss value;
    gre-in {
      mss value;
    }
    gre-out {
      mss value;
    }
    ipsec-vpn {
      mss value;
    }
  }
  tcp-session {
    fin-invalidate-session;
    no-sequence-check;
    no-syn-check;
    no-syn-check-in-tunnel;
    rst-invalidate-session;
    rst-sequence-check;
    strict-syn-check;
    tcp-initial-timeout seconds;
    time-wait-state {
```

```
        (session-ageout | session-timeout seconds);
      }
    }
    traceoptions {
      file {
        filename;
        files number;
        match regular-expression;
        size maximum-file-size;
        (world-readable | no-world-readable);
      }
      flag flag;
      no-remote-trace;
      packet-filter filter-name {
        destination-port port-identifier;
        destination-prefix address;
        interface interface-name;
        protocol protocol-identifier;
        source-port port-identifier;
        source-prefix address;
      }
      rate-limit messages-per-second;
    }
}
```

**Hierarchy Level** [edit security]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Determine how the device manages packet flow. The device can regulate packet flow in the following ways:

**Options** The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- *Juniper Networks Devices Processing Overview*
- *Understanding Session Characteristics for SRX Series Services Gateways*
- *Understanding Flow in Logical Systems for SRX Series Devices*

## forwarding-classes (CoS)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
forwarding-classes {
  class class-name {
    priority (high | low);
    queue-num number;
    spu-priority (high | low | medium-high | medium-low);
  }
  queue queue-number {
    class-name {
      priority (high | low);
    }
  }
}
```

**Hierarchy Level** [edit class-of-service]

**Release Information** Statement introduced in Junos OS Release 8.5. Statement updated in Junos OS Release 11.4. The **spu-priority** option introduced in Junos OS Release 11.4R2.

**Description** Configure forwarding classes and assign queue numbers.

**Options**

- **class class-name**—Display the forwarding class name assigned to the internal queue number.

  NOTE: This option is supported only on SRX1500, SRX5400, SRX5600, and SRX5800.

  NOTE: AppQoS forwarding classes must be different from those defined for interface-based rewriters.

- **priority**—Fabric priority value:
  - **high**—Forwarding class' fabric queuing has high priority.
  - **low**—Forwarding class' fabric queuing has low priority.

  The default **priority** is **low**.
- **queue queue-number**—Specify the internal queue number to which a forwarding class is assigned.
- **spu-priority**—Services Processing Unit (SPU) priority queue, **high**, **medium-high**, **medium-low**, or **low**. The default **spu-priority** is **low**.

NOTE: The spu-priority option is only supported on SRX1500 devices and SRX5000 line devices.

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

**Related Documentation**
- *Example: Configuring AppQoS*

## global-mac-table-aging-time (Protocols)

**Supported Platforms** SRX Series, vSRX

**Syntax** global-mac-table-aging-time *seconds*;

**Hierarchy Level** [edit protocols l2-learning]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Configure the timeout interval for entries in the MAC table.

**Default** 300 seconds

**Options**
***seconds***—Time elapsed before MAC table entries are timed out and entries are deleted from the table.
**Range:** 10 through 64,000 seconds

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

**Related Documentation**
- Example: Configuring VLANs on Security Devices on page 31

## global-mac-limit (Protocols)

**Supported Platforms** SRX Series

**Syntax**

```
global-mac-limit limit {
    packet-action drop;
}
```

**Hierarchy Level** [edit protocols l2-learning]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Limit the number of media access control (MAC) addresses learned from the logical interfaces on the router.

**Default** 131,071 MAC addresses

NOTE: SRX300, SRX320, SRX340, and SRX345 devices support 16,383 addresses, and SRX1500 devices support 24,575 addresses.

**Options** *limit*—Number of MAC addresses that can be learned on the device.

**Range:** 20 through 13,1071 addresses

The remaining statement is explained separately. See CLI Explorer.

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

**Related Documentation**

- Example: Configuring VLANs on Security Devices on page 31

## global-mode (Protocols)

**Supported Platforms** SRX Series, vSRX

**Syntax**
```
global-mode (switching | transparent-bridge) ;
```

**Hierarchy Level** [edit protocols l2-learning]

**Release Information** Statement introduced in Junos OS Release 15.1X49-D40.

**Description** Specify the global mode for the SRX Series device as Layer 2 transparent bridge mode or switching mode. After changing the mode, you must reboot the device for the configuration to take effect.

**Default** On SRX1500, the default Layer 2 global mode is transparent-bridge mode.

Starting with Junos OS Release 15.1X49-D100, on SRX300, SRX320, SRX340, SRX345, SRX550, and SRX550M devices, the default Layer 2 global mode configuration is changed from transparent-bridge to switching mode.

NOTE: You must explicitly configure Layer 2 transparent-bridge mode for the SRX300, SRX320, SRX340, SRX345, SRX550, and SRX550M devices that work in transparent mode. Use the command **set protocols l2-learning global-mode transparent-bridge** before rebooting the devices with Junos OS 15.1X49-D100 image.

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

**Related Documentation**

- l2-learning (Protocols) on page 348
- Ethernet Switching and Layer 2 Transparent Mode Overview on page 3

## global-no-mac-learning (Protocols)

**Supported Platforms** SRX Series

**Syntax** global-no-mac-learning;

**Hierarchy Level** [edit protocols l2-learning]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Disable MAC learning for the entire device.

**Default** MAC learning is enabled.

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

**Related Documentation**

- Example: Configuring VLANs on Security Devices on page 31

## host-inbound-traffic

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
host-inbound-traffic {
  protocols protocol-name {
    except;
  }
  system-services service-name {
    except;
  }
}
```

**Hierarchy Level** [edit security zones functional-zone management],
[edit security zones functional-zone management interfaces *interface-name*],
[edit security zones security-zone *zone-name*],
[edit security zones security-zone *zone-name* interfaces *interface-name*]

**Release Information** Statement introduced in Junos OS Release 8.5.

**Description** Control the type of traffic that can reach the device from interfaces bound to the zone.

**Options** The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- *Understanding How to Control Inbound Traffic Based on Traffic Types*
- *Understanding How to Control Inbound Traffic Based on Protocols*

## inet6 (Security Forwarding Options)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
inet6 {
  mode (drop | flow-based | packet-based);
}
```

**Hierarchy Level** [edit security forwarding-options family]

**Release Information** Statement introduced in Junos OS Release 8.5.

**Description** Enable packet-based or flow-based processing of IPv6 traffic. By default, the device drops IPv6 traffic.

NOTE: Packet-based processing is not supported on the following SRX Series devices: SRX1500, SRX5600, and SRX5800.

**Options** The **mode** statement is described separately.

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

**Related Documentation**

- family inet6 on page 331

## interfaces (CoS)

**Syntax**

```
interfaces
  interface-name {
    input-scheduler-map map-name ;
    input-shaping-rate rate ;
    scheduler-map map-name ;
    scheduler-map-chassis map-name ;
    shaping-rate rate ;
    unit logical-unit-number {
      adaptive-shaper adaptive-shaper-name ;
      classifiers {
        (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence)
        ( classifier-name | default);
      }
      forwarding-class class-name ;
      fragmentation-map map-name ;
      input-scheduler-map map-name ;
      input-shaping-rate (percent percentage | rate );
      input-traffic-control-profile profiler-name shared-instance instance-name ;
      loss-priority-maps {
        default;
        map-name ;
      }
      output-traffic-control-profile profile-name shared-instance instance-name ;
      rewrite-rules {
        dscp ( rewrite-name | default);
        dscp-ipv6 ( rewrite-name | default);
        exp ( rewrite-name | default) protocol protocol-types ;
        frame-relay-de ( rewrite-name | default);
        inet-precedence ( rewrite-name | default);
      }
      scheduler-map map-name ;
      shaping-rate rate ;
      virtual-channel-group group-name ;
    }
  }
}
```

**Hierarchy Level** [edit class-of-service interface *interface-name* unit *number*]

**Release Information** Statement introduced in Junos OS Release 8.5.

**Description** Associate the class-of-service configuration elements with an interface.

**Options** interface *interface-name* unit *number*—The user-specified interface name and unit number.

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

**Related Documentation**
- *Class of Service Feature Guide for Security Devices*

## interface (MVRP)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**

```
interface (all | interface-name) {
  join-timer milliseconds;
  leave-timer milliseconds;
  leaveall-timer milliseconds;
  point-to-point;
  registration (forbidden | normal | restricted);
}
```

**Hierarchy Level** [edit protocols mvrp],
[edit routing-instances *routing-instance-name* protocols mvrp] (for virtual switch instance type)

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Specify interfaces on which to configure Multiple VLAN Registration Protocol (MVRP).

**Default** By default, MVRP is disabled.

**Options**

**all**—Configure MVRP on all interfaces on the SRX Series device.

***interface-name***—Configure MVRP on specific interfaces.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**
- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## interfaces (Security Zones)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
interfaces interface-name {
  host-inbound-traffic {
    protocols protocol-name {
      except;
    }
    system-services service-name {
      except;
    }
  }
}
```

**Hierarchy Level** [edit security zones functional-zone management],
[edit security zones security-zone *zone-name*]

**Release Information** Statement introduced in Junos OS Release 8.5.

**Description** Specify the set of interfaces that are part of the zone.

**Options** ***interface-name*** —Name of the interface.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- *Understanding Security Zones*

## interface (Switching Options)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
interface interface-name {
  encapsulation-type;
  ignore-encapsulation-mismatch;
  pseudowire-status-tlv;
  static-mac mac-address {
    vlan-id vlan-id;
  }
}
```

**Hierarchy Level** [edit vlans *vlans-name* switch-options]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Specify the logical interfaces to include in the VLAN.

**Options**

- ***interface-name***—Name of a logical interface.
- ***encapsulation-type***—Encapsulation type for VPN.
- ***ignore-encapsulation-mismatch***—Allow different encapsulation types on local and remote devices.
- ***pseudowire-status-tlv***—Send pseudowire status.
- ***mac-address***—Static MAC address assigned to the logical interface.
- ***vlan-id***—VLAN identifier.

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

**Related Documentation**

- Understanding VLANs on Security Devices on page 29

## join-timer (MVRP)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**

```
join-timer milliseconds;
```

**Hierarchy Level**

```
[edit protocols mvrp interface (all | interface-name)],
[edit routing-instances routing-instance-name protocols mvrp] (for virtual switch instance type),
[edit routing-instances routing-instance-name protocols mvrp interface (all | interface-name)] (for virtual switch instance type)
```

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** For Multiple VLAN Registration Protocol (MVRP), configure the maximum interval interfaces must wait before sending MVRP protocol data units (PDUs).

**Options**

***milliseconds***—Interval that the interface must wait before sending MVRP PDUs (range from 100 milliseconds through 500 milliseconds). Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.

**Required Privilege Level**

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

**Related Documentation**

- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## l2-learning (Protocols)

**Supported Platforms** SRX Series

**Syntax**

```
l2-learning {
  global-mac-limit limit {
  packet-action-drop
  }
  global-mac-table-aging-time seconds;
  global-mode (switching | transparent-bridge) ;
  global-no-mac-learning;
}
```

**Hierarchy Level** [edit protocols]

**Release Information** Statement modified in Junos OS Release 9.5. Support for global mode added in Junos OS Release 15.1X49-D40.

**Description** Configure Layer 2 address learning and forwarding properties globally.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- global-mac-table-aging-time (Protocols) on page 337
- global-mac-limit (Protocols) on page 338
- global-no-mac-learning (Protocols) on page 340
- global-mode (Protocols) on page 339

## leave-timer (MVRP)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**
```
leave-timer milliseconds;
```

**Hierarchy Level**
```
[edit protocols mvrp interface (all | interface-name)],
[edit routing-instances routing-instance-name protocols mvrp] (for virtual switch instance type),
[edit routing-instances routing-instance-name protocols mvrp interface (all | interface-name)] (for virtual switch instance type)
```

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** For Multiple VLAN Registration Protocol (MVRP), configure the number of milliseconds the switch retains a VLAN in the Leave state before the VLAN is unregistered. If the interface receives a join message before this timer expires, the VLAN remains registered.

**Default** 1000 milliseconds

**Options**

***milliseconds***—Interval that the switch retains a VLAN in the Leave state before the VLAN is unregistered. At a minimum, set the **leave-timer** interval at twice the join-timer interval (range from 300 milliseconds through 1000 milliseconds). Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.

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

**Related Documentation**

- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## leaveall-timer (MVRP)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** leaveall-timer *milliseconds*;

**Hierarchy Level** [edit protocols mvrp interface (all | *interface-name*)],
[edit routing-instances *routing-instance-name* protocols mvrp] (for virtual switch instance type),
[edit routing-instances *routing-instance-name* protocols mvrp interface (all | *interface-name*)] (for virtual switch instance type)

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** For Multiple VLAN Registration Protocol (MVRP), configure the interval at which the LeaveAll state operates on the interface.

**Default** 60 seconds

**Options** ***seconds***—Interval between the sending of Leave All messages (range from 10 seconds through 60 seconds. Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.

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

**Related Documentation**

- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## loss-priority (CoS Loss Priority)

**Supported Platforms** SRX Series, vSRX

**Syntax** loss-priority *level* code-points [*values* ];

**Hierarchy Level** [edit class-of-service loss-priority-maps frame-relay-de *map-name*]

**Release Information** Statement introduced in Junos OS Release 9.2.

**Description** Map CoS values to a packet loss priority (PLP). In Junos OS, classifiers associate incoming packets with a forwarding class (FC) and PLP. PLPs allow you to set the priority for dropping packets. Typically, you mark packets exceeding some service level with a high loss priority—that is, a greater likelihood of being dropped.

**Options** ***level*** can be one of the following:

- **high**—Packet has high loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **medium-low**—Packet has medium-low loss priority.
- **low**—Packet has low loss priority.

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

**Related Documentation**

- *Understanding Interfaces*
- Understanding Packet Loss Priorities on page 197

## match (Security Policies)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
match {
  application {
    [application];
    any;
  }
  destination-address {
    [address];
    any;
    any-ipv4;
    any-ipv6;
  }
  source-address {
    [address];
    any;
    any-ipv4;
    any-ipv6;
  }
  source-identity {
    [role-name];
    any;
    authenticated-user;
    unauthenticated-user;
    unknown-user;
  }
}
```

**Hierarchy Level** [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name*]

**Release Information** Statement introduced in Junos OS Release 8.5. Statement updated with the **source-identity** option in Junos OS Release 12.1.

**Description** Configure security policy match criteria.

**Options** The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- *Security Policies Overview*

## mvrp

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**

```
mvrp {
  bpdu-destination-mac-address provider-bridge-group;
  join-timer milliseconds;
  leave-timer milliseconds;
  leaveall-timer milliseconds;
  interface (all | interface-name) {
    join-timer milliseconds;
    leave-timer milliseconds;
    leaveall-timer milliseconds;
    point-to-point;
    registration (forbidden | normal | restricted);
  }
  no-attribute-length-in-pdu
  no-dynamic-vlan;
  traceoptions {
    file filename <files number > <size size> <no-stamp | world-readable |
      no-world-readable>;
    flag flag;
  }
}
```

**Hierarchy Level** [edit protocols],
[edit routing-instances *routing-instance-name* protocols] (for virtual switch instance type),

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** For Layer 2 networks, configure Multiple VLAN Registration Protocol (MVRP) to dynamically share VLAN information and dynamically configure needed VLANs. Maintaining VLAN configurations based on active VLANs reduces the amount of traffic traveling in the network, saving network resources. MVRP is configured on trunk interfaces.

The remaining statements are explained separately. See CLI Explorer.

**Default** MVRP is disabled by default.

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

**Related Documentation**

- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## native-vlan-id (Interfaces)

**Supported Platforms** SRX Series, vSRX

**Syntax** native-vlan-id*vlan-id*;

**Hierarchy Level** [edit interfaces *interface-name* ]

**Release Information** Statement introduced in Junos OS Release 9.5.

**Description** Configure VLAN identifier for untagged packets received on the physical interface of a trunk mode interface.

**Options** ***vlan-id***—Configure a VLAN identifier for untagged packets. Enter a number from 0 through 4094.

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

**Related Documentation**

- *Understanding Interfaces*

## no-attribute-length-in-pdu

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** no-attribute-length-in-pdu;

**Hierarchy Level** [edit protocols mvrp]

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Include an extra byte in protocol data units (PDUs) sent by Multiple VLAN Registration Protocol (MVRP). You can recognize an MVRP version compatibility issue by observing the switch running the ELS version of MVRP. Because a switch running the ELS version of MVRP cannot interpret an unmodified PDU from a switch running the non-ELS version of MVRP, the switch does not add VLANs from the non-ELS version of MVRP. When you execute the command **show mvrp statistics** in the ELS version of MVRP, the values for **Received Join Empty** and **Received Join In** incorrectly display as zero, even though the value for the **Received MVRP PDUs without error** has been increased. Another indication that MVRP is having a version compatibility issue is that unexpected VLAN activity, such as multiple VLAN creation, taking place on the switch running the ELS version of MVRP.

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

**Related Documentation**

- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## no-dynamic-vlan

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**
```
no-dynamic-vlan;
```

**Hierarchy Level** [edit protocols mvrp],
[edit routing-instances *routing-instance-name* protocols mvrp] (for virtual switch instance type)

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Disable the dynamic creation of VLANs using Multiple VLAN Registration Protocol (MVRP) for interfaces participating in MVRP.

Dynamic VLAN configuration can be enabled on an interface independent of MVRP. The MVRP dynamic VLAN configuration setting does not override the interface configuration dynamic VLAN configuration setting. If dynamic VLAN creation is disabled, no dynamic VLANs are created on the interfaces, including dynamic VLANs created using MVRP.

This option can be applied only globally; it cannot be applied per interface.

**Default** If MVRP is enabled, the dynamic creation of VLANs as a result of MVRP protocol exchange messages is enabled.

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

**Related Documentation**
- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## peer-selection-service

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
peer-selection-service {
  command binary-file-path;
  disable;
  failover (alternate-media | other-routing-engine);
}
```

**Hierarchy Level** [edit system processes]

**Release Information** Statement introduced in Junos OS Release 8.5.

**Description** Enable the peer selection service process.

**Options**

- **command** ***binary-file-path***—Path to the binary process.
- **disable**—Disable the peer selection service process.
- **failover**—Configure the device to reboot if the software process fails four times within 30 seconds, and specify the software to use during the reboot.
  - **alternate-media**—Configure the device to switch to backup media that contains a version of the system if a software process fails repeatedly.
  - **other-routing-engine**—Instruct the secondary Routing Engine to take mastership if a software process fails. If this statement is configured for a process, and that process fails four times within 30 seconds, then the device reboots from the secondary Routing Engine.

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

**Related Documentation**

- Interfaces Feature Guide for Security Devices

## pgcp-service

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
pgcp-service {
  command binary-file-path;
  disable;
  failover (alternate-media | other-routing-engine);
}
```

**Hierarchy Level** [edit system processes]

**Release Information** Statement introduced in Junos OS Release 8.5.

**Description** Specify the Packet Gateway Control Protocol (PGCP) that is required for the border gateway function (BGF) feature.

**Options**

- **command *binary-file-path***—Path to the binary process.
- **disable**—Disable the Packet Gateway Control Protocol (PGCP) process.
- **failover**—Configure the device to reboot if the software process fails four times within 30 seconds, and specify the software to use during the reboot.
  - **alternate-media**—Configure the device to switch to backup media that contains a version of the system if a software process fails repeatedly.
  - **other-routing-engine**—Instruct the secondary Routing Engine to take mastership if a software process fails. If this statement is configured for a process, and that process fails four times within 30 seconds, the device reboots from the secondary Routing Engine.

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

**Related Documentation**

- *restart (Reset)*

## point-to-point (MVRP)

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**

```
point-to-point;
```

**Hierarchy Level**

```
[edit protocols mvrp interface (all | interface-name)],
[edit routing-instances routing-instance-name protocols mvrp interface (all | interface-name)]
  (for virtual switch instance type)
```

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** (Optional) For Multiple VLAN Registration Protocol (MVRP) configurations, configure an interface to be recognized as a point-to-point connection. If specified, a point-to-point subset of the MRP state machine is used to provide a simpler and more efficient method to accelerate convergence on the network. Point-to-point must be enabled after enabling MVRP for the interface to be recognized as a point-to-point connection.

**Default** MVRP is disabled by default.

**point-to-point** is disabled by default.

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

**Related Documentation**

- join-timer on page 347
- leave-timer on page 349
- leaveall-timer on page 350
- registration on page 367
- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## policy (Security Policies)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
policy policy-name {
  description description;
  match {
    application {
      [application];
      any;
    }
    destination-address {
      [address];
      any;
      any-ipv4;
      any-ipv6;
    }
    source-address {
      [address];
      any;
      any-ipv4;
      any-ipv6;
    }
    source-identity {
      [role-name];
      any;
      authenticated-user;
      unauthenticated-user;
      unknown-user;
    }
  }
  scheduler-name scheduler-name;
  then {
    count {
      alarm {
        per-minute-threshold number;
        per-second-threshold number;
      }
    }
    deny;
    log {
      session-close;
      session-init;
    }
    permit {
      application-services {
        application-firewall {
          rule-set rule-set-name;
        }
        application-traffic-control {
          rule-set rule-set-name;
        }
        gprs-gtp-profile profile-name;
        gprs-sctp-profile profile-name;
```

```
            idp;
            redirect-wx | reverse-redirect-wx;
            ssl-proxy {
              profile-name profile-name;
            }
            uac-policy {
              captive-portal captive-portal;
            }
            utm-policy policy-name;
          }
          destination-address {
            drop-translated;
            drop-untranslated;
          }
          firewall-authentication {
            pass-through {
              access-profile profile-name;
              client-match user-or-group-name;
              web-redirect;
            }
            user-firewall {
              access-profile profile-name;
              domain domain-name
              ssl-termination-profile profile-name;
            }
            web-authentication {
              client-match user-or-group-name;
            }
          }
          services-offload;
          tcp-options {
            initial-tcp-mss mss-value;
            reverse-tcp-mss mss-value;
            sequence-check-required;
            syn-check-required;
          }
          tunnel {
            ipsec-group-vpn group-vpn;
            ipsec-vpn vpn-name;
            pair-policy pair-policy;
          }
        }
        reject;
      }
    }
```

**Hierarchy Level** [edit security policies from-zone *zone-name* to-zone *zone-name*]

**Release Information** Statement introduced in Junos OS Release 8.5. The **services-offload** option added in Junos OS Release 11.4. Statement updated with the **source-identity** option and the **description** option added in Junos OS Release 12.1. Support for the **user-firewall** option added in Junos OS Release 12.1X45-D10. Support for the **initial-tcp-mss** and **reverse-tcp-mss** options added in Junos OS Release 12.3X48-D20.

**Description** Define a security policy.

**Options** ***policy-name***—Name of the security policy.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- *Configuring SSL Proxy*
- *Security Policies Overview*

## profile (Access)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
profile profile-name  {
  accounting {
    accounting-stop-on-access-deny;
    accounting-stop-on-failure;
    coa-immediate-update;
    duplication;
    immediate-update;
    order [accounting-method];
    statistics (time | volume-time);
    update-interval minutes;
  }
  accounting-order [accounting-method];
  address-assignment pool pool-name;
  authentication-order [ldap | none | password | securid];
  authorization-order [jsrc];
  client client-name  {
    chap-secret  chap-secret;
    client-group [ group-names ];
    firewall-user {
      password password;
    }
    no-rfc2486;
    pap-password pap-password;
    x-auth ip-address;
  }
  client-name-filter {
    count number;
    domain-name domain-name;
    separator special-character;
  }
  ldap-options {
    assemble {
      common-name common-name;
    }
    base-distinguished-name base-distinguished-name;
    revert-interval seconds;
    search {
      admin-search {
        distinguished-name distinguished-name;
        password password;
      }
      search-filter search-filter-name;
    }
  }
  ldap-server server-address {
    port port-number;
    retry attempts;
    routing-instance routing-instance-name;
    source-address source-address;
    timeout seconds;
```

```
  }
  provisioning-order (gx-plus | jsrc);
  service {
    accounting-order {
      activation-protocol;
      radius;
    }
  }
  session-options {
    client-group [group-name];
    client-idle-timeout minutes;
    client-session-timeout minutes;
  }
}
```

**Hierarchy Level** [edit access]

**Release Information** Statement introduced in Junos OS Release 10.4.

**Description** Create a profile containing a set of attributes that define device management access.

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

**Related Documentation**

- *Understanding Interfaces*
- *Understanding User Authentication for Security Devices*
- Ethernet Switching and Layer 2 Transparent Mode Overview on page 3

## recovery-timeout

**Supported Platforms** SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** recovery-timeout *seconds*;

**Hierarchy Level** [edit interfaces *interface-name* unit 0 family ethernet-switching]

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70.

**Description** Configure an interface to be temporarily disabled when MAC limiting is in effect with the action **shutdown**. This enables the affected interface to recover automatically from the error condition after the specified period of time:

- If you configure MAC limiting with the **shutdown** option and you enable **recovery-timeout**, the interface is temporarily disabled when the MAC address limit is reached. The interface will recover automatically after the number of seconds specified.

NOTE: The recovery-timeout configuration does not apply to preexisting error conditions. It impacts only error conditions that are detected after the recovery-timeout statement is configured and committed. To clear a preexisting error condition and restore the interface to service, use the operational mode commands clear ethernet-switching recovery-timeout .

**Default** The interface does not automatically recover from an error condition.

**Options** ***seconds***— Number of seconds that the interface remains in a disabled state due to a port error prior to automatic recovery.

**Range:** 10 through 3600

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

**Related Documentation**

- clear ethernet-switching recovery-timeout on page 394
- Understanding MAC Limiting on page 263
- Example: Configuring MAC Limiting on a Security Device on page 265
- Configuring Autorecovery From the Disabled State on Secure Interfaces of a Security Device (CLI Procedure) on page 267

## redundancy-group (Interfaces)

**Syntax** **redundancy-group** *number* ;

**Hierarchy Level** **[edit interfaces** *interface-name* **redundant-ether-options]**

**Release Information** Statement introduced in Junos OS Release 9.0.

**Description** Specify the redundancy group that a redundant Ethernet interface belongs to.

**Options** ***number*** —Number of the redundancy group that the redundant interface belongs to. Failover properties of the interface are inherited from the redundancy group.

**Range:** 1 through 255

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

**Related Documentation**

- Interfaces Feature Guide for Security Devices

## registration

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**
```
registration (forbidden | normal | restricted);
```

**Hierarchy Level**
```
[edit protocols mvrp interface (all | interface-name)],
[edit routing-instances routing-instance-name protocols mvrp interface (all | interface-name)]
    (for virtual switch instance type),
```

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** For Multiple VLAN Registration Protocol (MVRP) configurations, configure the registration mode for the interface.

**Default** **normal**—The interface or interfaces accept MVRP messages and participate in MVRP.

**Options** **forbidden**—The interface or interfaces do not register and do not participate in MVRP.

**restricted**—The interface or interfaces ignore all MVRP JOIN messages received for VLANs that are not statically configured for MVRP on the interface.

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

**Related Documentation**

- join-timer on page 347
- leave-timer on page 349
- leaveall-timer on page 350
- registration on page 367
- Understanding Multiple VLAN Registration Protocol (MVRP) for Dynamic VLAN Registration on Security Devices on page 125

## secure-wire

**Supported Platforms** SRX Series

**Syntax** secure-wire *secure-wire-name* interface [*interface-name-1 interface-name-2*];

**Hierarchy Level** [edit security forwarding-options]

**Release Information** Statement introduced in Junos OS Release 12.3X48-D10.

**Description** Configure mapping of interfaces through which traffic is forwarded unchanged.

**Options** **secure secure-wire**—Specify a name for the secure wire interface mapping.

**interface-name-1 interface-name-2**—Specify a pair of peer logical interfaces that constitutes the secure wire mapping.

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

**Related Documentation**

- Understanding Secure Wire on Security Devices on page 75

## security-zone

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
security-zone zone-name {
  address-book {
    address address-name {
      ip-prefix {
        description text;
      }
      description text;
      dns-name domain-name {
        ipv4-only;
        ipv6-only;
      }
      range-address lower-limit to upper-limit;
      wildcard-address ipv4-address/wildcard-mask;
    }
    address-set address-set-name {
      address address-name;
      address-set address-set-name;
      description text;
    }
  }
  advance-policy-based-routing;
  application-tracking;
  description text;
  enable-reverse-reroute;
  host-inbound-traffic {
    protocols protocol-name {
      except;
    }
    system-services service-name {
      except;
    }
  }
  interfaces interface-name {
    host-inbound-traffic {
      protocols protocol-name {
        except;
      }
      system-services service-name {
        except;
      }
    }
  }
  screen screen-name;
  tcp-rst;
}
```

**Hierarchy Level** [edit security zones]

**Release Information** Statement introduced in Junos OS Release 8.5. Support for wildcard addresses added in Junos OS Release 11.1. The **description** option added in Junos OS Release 12.1.

**Description** Define a security zone, which allows you to divide the network into different segments and apply different security options to each segment.

**Options** ***zone-name*** —Name of the security zone.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- *Security Zones and Interfaces Overview*
- *Example: Configuring Application Firewall Rule Sets Within a Security Policy*

## shaping-rate (CoS Interfaces)

**Supported Platforms** SRX Series, vSRX

**Syntax** shaping-rate *rate*;

**Hierarchy Level** [edit class-of-service interfaces *interface-name*],
[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]

**Release Information** Statement introduced in Junos OS Release 9.2.

**Description** For logical interfaces on which you configure packet scheduling, configure traffic shaping by specifying the amount of bandwidth to be allocated to the logical interface.

Logical and physical interface traffic shaping can be configured together. This means you can include the **shaping-rate** statement at the **[edit class-of-service interfaces interface *interface-name*]** hierarchy level *and* the **[edit class-of-service interfaces interface *interface-name* unit *logical-unit-number*]** hierarchy level. If you configure traffic shaping at both the logical and physical interface levels, the logical interface shaping credit is checked and updated before the physical interface shaping credit.

Alternatively, you can configure a shaping rate for a logical interface and oversubscribe the physical interface by including the **shaping-rate** statement at the **[edit class-of-service traffic-control-profiles]** hierarchy level. With this configuration approach, you can independently control the delay-buffer rate.

**Default** If you do not include this statement at the **[edit class-of-service interfaces interface *interface-name* unit *logical-unit-number*]** hierarchy level, the default logical interface bandwidth is the average of unused bandwidth for the number of logical interfaces that require default bandwidth treatment. If you do not include this statement at the **[edit class-of-service interfaces interface *interface-name*]** hierarchy level, the default physical interface bandwidth is the average of unused bandwidth for the number of physical interfaces that require default bandwidth treatment.

**Options** ***rate***—Peak rate, in bits per second (bps). You can specify a value in bits per second either as a complete decimal number or as a decimal number followed by the abbreviation **k** (1000), **m** (1,000,000), or **g** (1,000,000,000).

**Range:** For logical interfaces, 1000 through 6,400,000,000,000 bps.

For physical interfaces, 1000 through 6,400,000,000,000 bps.

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

**Related Documentation**

- Class of Service Feature Guide for Security Devices

## source-address (Security Policies)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
source-address {
  [address];
  any;
  any-ipv4;
  any-ipv6;
}
```

**Hierarchy Level** [edit security policies from-zone *zone-name* to-zone *zone-name* policy *policy-name* match]

[edit security policies global policy *policy-name* match]

**Release Information** Statement introduced in Junos OS Release 8.5. Support for IPv6 addresses added in Junos OS Release 10.2. Support for IPv6 addresses in active/active chassis cluster configurations (in addition to the existing support of active/passive chassis cluster configurations) added in Junos OS Release 10.4. Support for wildcard addresses added in Junos OS Release 11.1.

**Description** Define the matching criteria. You can specify one or more IP addresses, address sets, or wildcard addresses. You can specify wildcards **any**, **any-ipv4**, or **any-ipv6**.

**Options** ***address***—IP addresses, address sets, or wildcard addresses (represented as A.B.C.D/wildcard-mask). You can configure multiple addresses or address prefixes separated by spaces and enclosed in square brackets.

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

**Related Documentation**

- *Security Policies Overview*
- *Understanding Security Policy Rules*
- *Understanding Security Policy Elements*

## static-mac (VLANs)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
static-mac mac-address {
    vlan-id vlan-id;
}
```

**Hierarchy Level** [edit vlans*vlan--name* switch-options interface *interface-name*]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Configure a static MAC address for a logical interface in a VLAN.

**Options**

- ***mac-address***—MAC address
- ***vlan-id***—VLAN identifier

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

**Related Documentation**

- Understanding VLANs on Security Devices on page 29

## switch-options (VLANs)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
switch-options {
  interface interface-name {
    encapsulation-type;
    ignore-encapsulation-mismatch;
    pseudowire-status-tlv;
    static-mac mac-address {
      vlan-id vlan-id;
    }
  }
  mac-table-aging-time seconds;
  mac-table-size {
    number;
    packet-action drop;
  }
}
```

**Hierarchy Level** [edit vlans *vlans-name*]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Configure Layer 2 learning and forwarding properties for a VLAN.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- Ethernet Switching and Layer 2 Transparent Mode Overview on page 3

## system-services (Security Zones Interfaces)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
system-services service-name {
  except;
}
```

**Hierarchy Level** [edit security zones security-zone *zone-name* interfaces *interface-name* host-inbound-traffic]

**Release Information** Statement introduced in Junos OS Release 8.5.

**Description** Specify the types of traffic that can reach the device on a particular interface.

**Options**

- *service-name* —Service for which traffic is allowed. The following services are supported:
  - **all**—Enable all possible system services available on the Routing Engine (RE).
  - **any-service**—Enable services on entire port range.
  - **bootp**—Enable traffic destined to BOOTP and DHCP relay agents.
  - **dhcp**—Enable incoming DHCP requests.
  - **dhcpv6**—Enable incoming DHCP requests for IPv6.
  - **dns**—Enable incoming DNS services.
  - **finger**—Enable incoming finger traffic.
  - **ftp**—Enable incoming FTP traffic.
  - **http**—Enable incoming J-Web or clear-text Web authentication traffic.
  - **https**—Enable incoming J-Web or Web authentication traffic over Secure Sockets Layer (SSL).
  - **ident-reset**—Enable the access that has been blocked by an unacknowledged identification request.
  - **ike**—Enable Internet Key Exchange traffic.
  - **netconf SSH**—Enable incoming NetScreen Security Manager (NSM) traffic over SSH.
  - **ntp**—Enable incoming Network Time Protocol (NTP) traffic.
  - **ping**—Allow the device to respond to ICMP echo requests.
  - **r2cp**—Enable incoming Radio Router Control Protocol traffic.
  - **reverse-ssh**—Reverse SSH traffic.
  - **reverse-telnet**—Reverse Telnet traffic.
  - **rlogin**—Enable incoming **rlogin** (remote login) traffic.
  - **rpm**—Enable incoming real-time performance monitoring (RPM) traffic.

- **rsh**—Enable incoming Remote Shell (**rsh**) traffic.
- **snmp**—Enable incoming SNMP traffic (UDP port 161).
- **snmp-trap**—Enable incoming SNMP traps (UDP port 162).
- **ssh**—Enable incoming SSH traffic.
- **telnet**—Enable incoming Telnet traffic.
- **tftp**—Enable TFTP services.
- **traceroute**—Enable incoming traceroute traffic (UDP port 33434).
- **xnm-clear-text**—Enable incoming Junos XML protocol traffic for all specified interfaces.
- **xnm-ssl**— Enable incoming Junos XML protocol-over-SSL traffic for all specified interfaces.

- **except**—(Optional) except can only be used if all has been defined.

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

**Related Documentation**
- *Security Zones and Interfaces Overview*
- *Supported System Services for Host Inbound Traffic*

## unframed | no-unframed (Interfaces)

**Supported Platforms** SRX1500, SRX550M, vSRX

**Syntax**
```
(unframed | no-unframed);
```

**Hierarchy Level** [edit interfaces *interface-name* t3-options]

**Release Information** Statement introduced in Junos OS Release 11.1.

**Description** Enable or disable framing for the T3 interface on a 1-Port Clear Channel DS3/E3 GPIM on an SRX Series device. By default, unframed mode is enabled. Select no-unframed to enable framing. Select unframed to return to the default mode.

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

**Related Documentation**
- *Example: Configuring a T3 Interface*

## vlan-id (VLAN)

**Supported Platforms** EX Series, MX Series, SRX Series, vSRX

**Syntax** vlan-id (all | none | *number*);

**Hierarchy Level**
[edit vlans *vlan-name*],
[edit logical-systems *logical-system-name* vlans*vlan-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* vlans*vlan-name*],
[edit routing-instances *routing-instance-name* vlans *vlan--name*]

**Release Information**
Statement introduced in Junos OS Release 8.4.
Support for Layer 2 trunk ports added in Junos OS Release 9.2.
Support for SRX 5600, and SRX 5800 devices added in Junos OS Release 9.6.
Support for logical systems added in Junos OS Release 9.6.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

**Description** Specify a VLAN identifier (VID) to include in the packets sent to and from the 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 VLANs with a valid VLAN identifier, you must specify a unique VLAN identifier for each. However, you can use the same VLAN identifier for 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 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 VLAN.

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

NOTE: Multichassis link aggregation (MC-LAG) does not support the **none** option with the **vlan-id** statement with VLANs.

**Required Privilege Level**

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

**Related Documentation**

- Example: Configuring VLANs on Security Devices on page 31
- *Example: Configuring Interfaces and Routing Instances for a User Logical System*

## vlan-id-range

**Supported Platforms** SRX Series

**Syntax** vlan-id-range *vlan-id–vlan-id*

**Hierarchy Level** [edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

**Release Information** Statement introduced in Junos OS Release 8.4.

**Description** Bind a range of VLAN IDs to a logical interface.

**Options** **number**—The first number is the lowest VLAN ID in the range the second number is the highest VLAN ID in the range.

**Range:** 1 through 4094

NOTE: On SRX240, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX650 devices, the VLAN range from 1 to 4094 on inet interfaces and the VLAN range from 1 to 3967 on Ethernet switching interfaces. On Ethernet switching interfaces, the VLAN range from 3968 to 4094 falls under the reserved VLAN address range, and the user is not allowed to configure VLANs in this range.

NOTE: Configuring **vlan-id-range** with the entire vlan-id range is an unnecessary waste of system resources and is not best practice. It should be used only when a subset of VLAN IDs (not the entire range) needs to be associated with a logical interface. If you specify the entire range (1-4094), it has the same result as not specifying a range; however, it consumes PFE resources such as VLAN lookup tables entries, and so on.

The following examples illustrate this further:

```
[edit interfaces interface-name]
vlan-tagging;
unit number {
    vlan-id-range 1-4094;
}

[edit interfaces interface-name]
unit 0;
```

VLAN ID 0 is reserved for tagging the priority of frames.

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

**Related Documentation**
- Understanding VLANs on page 115

## vlan members (VLANs)

**Supported Platforms** SRX Series

**Syntax** vlan members [*vlan-id*];

**Hierarchy Level** [edit vlans *vlan-name*]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Specify multiple VLAN identifiers to create a VLAN for each VLAN identifier.

**Options** ***vlan-id***—A list of valid VLAN identifiers. A VLAN is created for each VLAN identifier in the list.

NOTE: If you specify a VLAN identifier list, you cannot configure an IRB interface in the VLAN.

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

**Related Documentation**
- Example: Configuring VLANs on Security Devices on page 31

## vlan-tagging (Interfaces)

**Supported Platforms** SRX Series, vSRX

**Syntax** vlan-tagging native-vlan-id *vlan-id*;

**Hierarchy Level** [edit interfaces *interface* ]

**Release Information** Statement introduced in Junos OS Release 9.5.

**Description** Configure VLAN identifier for untagged packets received on the physical interface of a trunk mode interface.

**Options** **native-vlan-id**—Configures a VLAN identifier for untagged packets. Enter a number from 0 through 4094.

NOTE: The native-vlan-id can be configured only when either flexible-vlan-tagging mode or interface-mode trunk is configured.

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

**Related Documentation**

- *Configuring VLAN Tagging*

## vlans

**Supported Platforms** SRX Series, vSRX

**List of Syntax**
Syntax (SRX Series) on page 382
Syntax (vSRX) on page 385

**Syntax (SRX Series)**

```
vlans {
  vlan name {
    (vlan-id (all | none | number) | vlan-id-list [ vlan-id-numbers] | vlan-tags <inner number>
      outer number);
    description text-description;
    forwarding-options {
      dhcp-security {
        arp-inspection;
        dhcpv6-options {
          option-16 {
            use-string use-string;
          }
          option-18 {
            prefix {
              host-name;
              logical-system-name;
              routing-instance-name;
              vlan-id;
              vlan-name;
            }
            use-interface-description (device | logical);
            use-interface-index (device | logical);
            use-interface-mac;
            use-interface-name (device | logical);
            use-string use-string;
          }
          option-37 {
            prefix {
              host-name;
              logical-system-name;
              routing-instance-name;
              vlan-id;
              vlan-name;
            }
            use-interface-description (device | logical);
            use-interface-index (device | logical);
            use-interface-mac;
            use-interface-name (device | logical);
            use-string use-string;
          }
        }
        group group-name {
          interface interface-name {
            static-ip {
              ip-address {
              mac-address;
              }
```

```
            }
            static-ipv6 {
              ip-address {
              mac-address;
              }
            }
          }
          overrides {
            no-dhcpv6-options;
            no-option16;
            no-option18;
            no-option37;
            no-option82;
            trusted;
            untrusted;
          }
        }
        ip-source-guard;
        ipv6-source-guard;
        light-weight-dhcpv6-relay;
        neighbor-discovery-inspection;
        no-dhcp-snooping;
        no-dhcpv6-snooping;
        option-82 {
          circuit-id {
            prefix {
              host-name;
              logical-system-name;
              routing-instance-name;
            }
            use-interface-description (device | logical);
            use-vlan-id;
          }
          remote-id {
            host-name;
            mac;
            use-interface-description (device | logical);
            use-string use-string;
          }
          vendor-id {
            use-string use-string;
          }
        }
      }
      filter {
        input filter-name;
      }
      flood {
        input filter-name;
      }
    }
    interface interface-name;
    l3-interface l3-interface-name;
    mcae-mac-synchronize;
    no-irb-layer-2-copy;
    service-id service-id;
```

```
switch-options {
  interface name {
    action-priority action-priority;
    encapsulation-type (ethernet | ethernet-vlan);
    ignore-encapsulation-mismatch;
    interface-mac-ip-limit limit;
    interface-mac-limit {
      limit;
      packet-action (drop | drop-and-log | log | none | shutdown);
    }
    mac-pinning;
    no-mac-learning;
    pseudowire-status-tlv;
    static-mac mac-address {
      vlan-id value;
    }
  }
  interface-mac-ip-limit limit;
  interface-mac-limit limit {
    packet-action (drop | drop-and-log | log | none | shutdown);
  }
  mac-ip-table-size limit;
  mac-statistics;
  mac-table-aging-time seconds;
  mac-table-size {
    limit;
    packet-action {
      drop;
    }
  }
  no-mac-learning;
  static-rvtep-mac {
    mac mac_addr {
      remote-vtep vtep;
    }
  }
 }
}
}
```

**Syntax (vSRX)**

```
vlans {
  vlan name {
    (vlan-id (all | none | number) | vlan-id-list [ vlan-id-numbers ] | vlan-tags <inner number>
      outer number);
    description text-description;
    forwarding-options {
      dhcp-security {
        arp-inspection;
        dhcpv6-options {
          option-16 {
            use-string use-string;
          }
          option-18 {
            prefix {
              host-name;
              logical-system-name;
              routing-instance-name;
              vlan-id;
              vlan-name;
            }
            use-interface-description (device | logical);
            use-interface-index (device | logical);
            use-interface-mac;
            use-interface-name (device | logical);
            use-string use-string;
          }
          option-37 {
            prefix {
              host-name;
              logical-system-name;
              routing-instance-name;
              vlan-id;
              vlan-name;
            }
            use-interface-description (device | logical);
            use-interface-index (device | logical);
            use-interface-mac;
            use-interface-name (device | logical);
            use-string use-string;
          }
        }
        group group-name {
          interface interface-name {
            static-ip {
              ip-address;
            }
            static-ipv6 {
              ip-address;
            }
          }
          overrides {
            no-dhcpv6-options;
            no-option16;
            no-option18;
            no-option37;
            no-option82;
```

```
            trusted;
            untrusted;
          }
        }
        ip-source-guard;
        ipv6-source-guard;
        light-weight-dhcpv6-relay;
        neighbor-discovery-inspection;
        no-dhcp-snooping;
        no-dhcpv6-snooping;
        option-82 {
          circuit-id {
            prefix {
              host-name;
              logical-system-name;
              routing-instance-name;
            }
            use-interface-description (device | logical);
            use-vlan-id;
          }
          remote-id {
            host-name;
            mac;
            use-interface-description (device | logical);
            use-string use-string;
          }
          vendor-id {
            use-string use-string;
          }
        }
      }
      filter {
        input filter-name;
      }
      flood {
        input filter-name;
      }
    }
    interface interface-name;
    l3-interface l3-interface-name;
    mcae-mac-synchronize;
    no-irb-layer-2-copy;
    service-id service-id;
    switch-options {
      interface name {
        action-priority action-priority;
        encapsulation-type (ethernet | ethernet-vlan);
        ignore-encapsulation-mismatch;
        interface-mac-limit {
          limit;
          packet-action (drop | drop-and-log | log | none | shutdown);
        }
        mac-pinning;
        no-mac-learning;
        pseudowire-status-tlv;
        static-mac mac-address {
```

```
            vlan-id value;
          }
        }
        interface-mac-ip-limit limit;
        interface-mac-limit limit {
          packet-action (none | shutdown);
        }
        mac-ip-table-aging-time seconds;
        mac-ip-table-size limit;
        mac-statistics;
        mac-table-aging-time seconds;
        mac-table-size {
          limit;
          packet-action {
            drop;
          }
        }
        no-mac-learning;
        static-rvtep-mac {
          mac mac_addr {
            remote-vtep vtep;
          }
        }
      }
    }
  }
}
```

**Hierarchy Level** [edit]

**Release Information** Statement introduced in Junos OS Release 15.1X49-D10.

**Description** Configure VLAN properties on SRX Series devices (including vSRX). The following configuration guidelines apply:

- Only private VLAN (PVLAN) firewall filters can be used when the VLAN is enabled for Q-in-Q tunneling.
- An S-VLAN tag is added to the packet if the VLAN is Q-in-Q-tunneled and the packet is arriving from an access interface.
- You cannot use a firewall filter to assign an integrated routing and bridging (IRB) interface or a routed VLAN interface (RVI) to a VLAN.
- VLAN assignments performed using a firewall filter override all other VLAN assignments.

**Options** ***vlan-name***—Name of the VLAN. The name can include letters, numbers, hyphens (-), and periods (.) and can contain up to 255 characters long.

The remaining statements are explained separately. See CLI Explorer.

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

**Related Documentation**

- Example: Configuring VLANs on Security Devices (CLI Procedure) on page 118
- Configuring Q-in-Q Tunneling on Security Devices on page 137

CHAPTER 25

# Operational Commands

- clear dot1x
- clear error bpdu interface
- clear ethernet-switching recovery-timeout
- clear mvrp statistics
- clear interfaces statistics swfabx
- clear oam ethernet connectivity-fault-management path-database
- clear oam ethernet connectivity-fault-management statistics
- clear security flow ip-action
- clear security flow session family
- show chassis cluster ethernet-switching interfaces
- show chassis cluster ethernet-switching status
- show chassis cluster status
- show dot1x authentication-bypassed-users
- show dot1x authentication-failed-users
- show dot1x interface
- show dot1x static-mac-address
- show dot1x statistics
- show ethernet-switching mac-learning-log (View)
- show ethernet-switching table (View)
- show interfaces (SRX Series)
- show interfaces swfabx
- show mvrp
- show mvrp applicant-state
- show mvrp dynamic-vlan-memberships
- show mvrp interface
- show mvrp registration-state
- show mvrp statistics
- show oam ethernet connectivity-fault-management adjacencies

- show oam ethernet connectivity-fault-management forwarding-state
- show oam ethernet connectivity-fault-management interfaces
- show oam ethernet connectivity-fault-management mep-database
- show oam ethernet connectivity-fault-management mep-statistics
- show oam ethernet connectivity-fault-management mip
- show oam ethernet connectivity-fault-management path-database
- show oam ethernet link-fault-management
- show security flow gate family
- show security flow ip-action
- show security flow session family
- show security flow statistics
- show security flow status
- show security forward-options secure-wire
- show security policies
- show security zones
- show spanning-tree interface
- show vlans

## clear dot1x

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**
clear dot1x
interface <*interface-name*>
mac-address <*static-mac-address*>
statistics <interface *interface-name*>

**Release Information** Command introduced in Junos OS Release 15.1X49-D80.

**Description** Reset the authentication state of an interface or delete 802.1X statistics from the device. When you reset an interface using the **interface** or **mac-address** options, reauthentication on the interface is also triggered. The device sends out a multicast message on the interface to restart the authentication of all connected supplicants. If a MAC address is reset, then the device sends out a unicast message to that specific MAC address to restart authentication.

**Options**

**interface <[*interface-name*]>**—Reset the authentication state of all the supplicants (also, clear all the authentication bypassed clients) connected to the specified interface (when the interface is an authenticator) or reset the authentication state for the interface itself (when the interface is a supplicant).

**mac-address [*mac-addresses*]**—Reset the authentication state of the specified MAC addresses.

**statistics <interface *interface-name*>**—Clear 802.1X statistics on all 802.1X-enabled interfaces. If the **interface** option is specified, clear 802.1X firewall statistics for that interface or interfaces.

**Required Privilege Level** view

**Related Documentation**
- dot1x on page 321

**List of Sample Output**
clear dot1x interface on page 391
clear dot1x mac-address on page 391
clear dot1x statistics interface on page 392

### Sample Output

#### clear dot1x interface

```
user@host> clear dot1x interface ge-0/0/1
```

#### clear dot1x mac-address

```
user@host> clear dot1x mac-address 00:04:ae:cd:23:5f
```

### clear dot1x statistics interface

```
user@host> clear dot1x statistics interface ge-0/0/1
```

## clear error bpdu interface

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** **clear error bpdu interface (all | *interface-name*)**

**Release Information** Command introduced in Junos OS Release 15.1X49-D70.

**Description** Clear a bridge protocol data unit (BPDU) error condition caused by the detection of a possible bridging loop from Spanning Tree Protocol (STP) operation.

**Required Privilege Level** clear

**Related Documentation**

- Understanding BPDU Protection for STP, RSTP, and MSTP on page 155
- Understanding Root Protection for STP, RSTP, and MSTP on page 171
- **disable-timeout (Spanning Trees) on page 319**

## clear ethernet-switching recovery-timeout

**Supported Platforms** SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**
clear ethernet-switching recovery-timeout
<interface *interface-name*>

**Release Information** Command introduced in Junos OS Release 15.1X49-D70

**Description** Clear all MAC limiting errors from all the Ethernet switching interfaces on the device or from the specified interface, and restore the interfaces or the specified interface to service.

**Options**
**interface *interface-name***—(Optional) Clear all MAC limiting errors from the specified interface and restore the interface to service.

**Required Privilege Level** clear

**Related Documentation**
- Understanding MAC Limiting on page 263
- Example: Configuring MAC Limiting on a Security Device on page 265
- Configuring Autorecovery From the Disabled State on Secure Interfaces of a Security Device (CLI Procedure) on page 267

## clear mvrp statistics

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax**
```
clear mvrp statistics
<interface interface-name>
<routing-instance routing-instance-name>
```

**Release Information** Statement introduced in Junos OS Release 15.1X49-D70.

**Description** Clear all Multiple VLAN Registration Protocol (MVRP) interface and routing instances statistics.

**Options**
- **interface**—Clear the MVRP interface statistics on the specified interface name.
- **routing-instances**— Clear the MVRP statistics on the specified routing instance. If you do not specify a routing instance name, statistics are cleared for the default routing instance.

**Required Privilege Level** clear

**Related Documentation**
- show mvrp on page 460

**Output Fields** This command produces no output.

## clear interfaces statistics swfabx

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M, vSRX

**Syntax** clear interfaces statistics <swfab0 | swfab1>

**Release Information** Command introduced in Junos OS Release 11.1.

**Description** Clear interface statistics for the specified swfab interface.

**Required Privilege Level** clear

**Related Documentation**
- show interfaces swfabx on page 458

**List of Sample Output** clear interfaces statistics <swfab0 | swfab1> on page 396

**Output Fields** When you enter this command, interface statistics for swfab0 and swfab1 are cleared.

### Sample Output

clear interfaces statistics <swfab0 | swfab1>

```
user@host> clear interfaces statistics <swfab0 | swfab1>
```

## clear oam ethernet connectivity-fault-management path-database

**Supported Platforms** SRX Series

**Syntax** clear oam ethernet connectivity-fault-management path-database maintenance-domain *md-name* maintenance-association *ma-name* host <*mac-addr*>

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Clear the relevant path information from the database for the specified remote host.

**Options** **host**—MAC address of remote host in xx:xx:xx:xx:xx:xx format.

**maintenance-association** —Name of the maintenance association.

**maintenance-domain** —Name of the maintenance domain.

**Required Privilege Level** clear

**Related Documentation**
- show oam ethernet connectivity-fault-management path-database on page 487

**List of Sample Output** clear oam ethernet connectivity-fault- management path-database on page 397

### Sample Output

#### clear oam ethernet connectivity-fault- management path-database

```
user@host> clear oam ethernet connectivity-fault-management path-database
maintenance-domain private maintenance-association private-ma 00:00:5E:00:53:AA
Path database entries cleared for the remote-host
```

## clear oam ethernet connectivity-fault-management statistics

**Supported Platforms** SRX Series

**Syntax**
clear oam ethernet connectivity-fault-management statistics
interface
level

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Clear connectivity fault management (CFM) statistics.

**Options**
**Interface**—Clear the statistics on an interface.

**Level**—The maintenance-domain level (0 through 7).

**Required Privilege Level** View

**Related Documentation**
- show oam ethernet connectivity-fault-management mep-statistics on page 482

**List of Sample Output** clear oam ethernet connectivity-fault- management statistics on page 398

**Output Fields** When you enter this command, you are provided feedback on the status of your request.

### Sample Output

#### clear oam ethernet connectivity-fault- management statistics

```
user@host> clear oam ethernet connectivity-fault-management statistics
Cleared statistics of all CFM sessions
```

## clear security flow ip-action

**Supported Platforms** SRX Series, vSRX

**Syntax** **clear security flow ip-action** [*filter*]

**Release Information** Command introduced in Junos OS Release 10.4. Logical systems option introduced in Junos OS Release 11.2.

**Description** Clear IP-action entries, based on filtered options, for IP sessions running on the device.

**Options** *filter*—Filter the display based on the specified criteria.

The following filters display those sessions that match the criteria specified by the filter. Refer to the sample output for filtered output examples.

**all |** [*filter*]—All active sessions on the device.

**destination-port** ***destination-port***—Destination port number of the traffic. Range is 1 through 65,535.

**destination-prefix** ***destination-prefix***—Destination IP prefix or address.

**family (inet | inet6)** [*filter*]—IPv4 traffic or IPv6-NATPT traffic and filtered options.

**logical-system** ***logical-system-name*** **| all** [*filter*]—Specified logical system or all logical systems.

**protocol** ***protocol-name*** **|** ***protocol-number*** [*filter*]—Protocol name or number and filtered options.

- **ah** or **51**
- **egp** or **8**
- **esp** or **50**
- **gre** or **47**
- **icmp** or **1**
- **icmp6** or **58**
- **ipip** or **4**
- **ospf** or **89**
- **pim** or **103**
- **rsvp** or **46**
- **sctp** or **132**
- **tcp** or **6**
- **udp** or **17**

**root-logical-system** [*filter*]—Default logical system information and filtered options.

**source-port** ***source-port***—Source port number of the traffic. Range is 1 through 65,535.

**source-prefix** ***source-prefix***—Source IP prefix or address of the traffic.

**Required Privilege Level** clear

**Related Documentation**
- show security flow ip-action on page 496

**List of Sample Output**
clear security flow ip-action all on page 400
clear security flow ip-action destination-prefix on page 400
clear security flow ip-action family inet on page 400
clear security flow ip-action protocol udp on page 400

**Output Fields** When you enter this command, the system responds with the status of your request.

## Sample Output

### clear security flow ip-action all

```
user@host>clear security flow ip-action all
1008 ip-action entries cleared
```

### clear security flow ip-action destination-prefix

```
user@host>clear security flow ip-action destination-prefix 192.0.2.5/24
87 ip-action entries cleared
```

### clear security flow ip-action family inet

```
user@host>clear security flow ip-action family inet
2479 ip-action entries cleared
```

### clear security flow ip-action protocol udp

```
user@host>clear security flow ip-action protocol udp
270 ip-action entries cleared
```

## clear security flow session family

**Supported Platforms** SRX Series, vSRX

**Syntax** clear security flow session family (inet | inet6)

**Release Information** Command introduced in Junos OS Release 10.2.

**Description** Clear sessions that match the specified protocol family.

**Options**
- **inet**—Clear IPv4 sessions.
- **inet6**—Clear IPv6 sessions.

**Required Privilege Level** clear

**Related Documentation**
- show security flow session family on page 504

**List of Sample Output**
clear security flow session family inet on page 401
clear security flow session family inet6 on page 401

**Output Fields** When you enter this command, you are provided feedback on the status of your request.

### Sample Output

#### clear security flow session family inet

```
user@host> clear security flow session family inet
1 active sessions cleared
```

#### clear security flow session family inet6

```
user@host> clear security flow session family inet6
1 active sessions cleared
```

## show chassis cluster ethernet-switching interfaces

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M, vSRX

**Syntax** show chassis cluster ethernet-switching interfaces

**Release Information** Command introduced in Junos OS Release 11.1.

**Description** Display the status of the switch fabric interfaces (swfab interfaces) in a chassis cluster.

**Required Privilege Level** view

**Related Documentation**
- *cluster (Chassis)*
- *Ethernet Switching and Layer 2 Transparent Mode Feature Guide for Security Devices*

**List of Sample Output** show chassis cluster ethernet-switching interfaces on page 402

**Output Fields** Table 38 on page 402 lists the output fields for the **show chassis cluster ethernet-switching interfaces** command. Output fields are listed in the approximate order in which they appear.

Table 38: show chassis cluster ethernet-switching interfaces Output Fields

| Field Name | Field Description |
|---|---|
| **swfab** ***switch fabric interface-name*** | Name of the switch fabric interface.<br>• Name—Name of the physical interface.<br>• Status—State of the switch fabric interface: up or down. |

## Sample Output

### show chassis cluster ethernet-switching interfaces

```
user@host> show chassis cluster ethernet-switching interfaces
swfab0:
    Name                Status
    ge-0/0/9            up
    ge-0/0/10           up
swfab1:
    Name                Status
    ge-7/0/9            up
    ge-7/0/10           up
```

## show chassis cluster ethernet-switching status

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M, vSRX

**Syntax** show chassis cluster ethernet-switching status

**Release Information** Command introduced in Junos OS Release 11.1.

**Description** Display the Ethernet switching status of the chassis cluster.

**Required Privilege Level** view

**Related Documentation**

- *cluster (Chassis)*
- *Ethernet Switching and Layer 2 Transparent Mode Feature Guide for Security Devices*

**List of Sample Output** show chassis cluster ethernet-switching status on page 404

**Output Fields** Table 39 on page 403 lists the output fields for the **show chassis cluster ethernet-switching status** command. Output fields are listed in the approximate order in which they appear.

### Table 39: show chassis cluster ethernet-switching status Output Fields

| Field Name | Field Description |
|---|---|
| **Cluster ID** | ID number (1-255) of a cluster. Setting a cluster ID to 0 is equivalent to disabling a cluster. A cluster ID greater than 15 can only be set when the fabric and control link interfaces are connected back-to-back. |
| **Redundancy-Group** | You can create up to 128 redundancy groups in the chassis cluster. |
| **Node name** | Node (device) in the chassis cluster (**node0** or **node1**). |
| **Priority** | Assigned priority for the redundancy group on that node. |
| **Status** | State of the redundancy group (**Primary**, **Secondary**, **Lost**, or **Unavailable**).<br>• **Primary**—Redundancy group is active and passing traffic.<br>• **Secondary**—Redundancy group is passive and not passing traffic.<br>• **Lost**—Node loses contact with the other node through the control link. Most likely to occur when both nodes are in a cluster and there is a control link failure, one node cannot exchange heartbeats, or when the other node is rebooted.<br>• **Unavailable**—Node has not received a single heartbeat over the control link from the other node since the other node booted up. Most likely to occur when one node boots up before the other node, or if only one node is present in the cluster. |

Table 39: show chassis cluster ethernet-switching status Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| **Preempt** | • **Yes**: Mastership can be preempted based on priority.<br>• **No**: Change in priority will not preempt mastership. |
| **Manual failover** | • **Yes**: Mastership is set manually through the CLI.<br>• **No**: Mastership is not set manually through the CLI. |

## Sample Output

### show chassis cluster ethernet-switching status

```
user@host> show chassis cluster ethernet-switching status

Monitor Failure codes:
    CS  Cold Sync monitoring        FL  Fabric Connection monitoring
    GR  GRES monitoring             HW  Hardware monitoring
    IF  Interface monitoring        IP  IP monitoring
    LB  Loopback monitoring         MB  Mbuf monitoring
    NH  Nexthop monitoring          NP  NPC monitoring
    SP  SPU monitoring              SM  Schedule monitoring
    CF  Config Sync monitoring

Cluster ID: 1
Node   Priority Status         Preempt Manual   Monitor-failures

Redundancy group: 0 , Failover count: 0
node0  1      primary        no      no       None
node1  1      secondary      no      no       None

Ethernet switching status:
    Probe state is UP. Both nodes are in single ethernet switching domain(s).
```

## show chassis cluster status

**Supported Platforms** SRX Series, vSRX

**Syntax**
**show chassis cluster status**
**<redundancy-group *group-number* >**

**Release Information** Support for monitoring failures added in Junos OS Release 12.1X47-D10.

**Description** Display the current status of the Chassis Cluster. You can use this command to check the status of chassis cluster nodes, redundancy groups, and failover status.

**Options**
- none—Display the status of all redundancy groups in the chassis cluster.
- **redundancy-group *group-number***—(Optional) Display the status of the specified redundancy group.

**Required Privilege Level** view

**Related Documentation**
- *redundancy-group (Chassis Cluster)*
- *clear chassis cluster failover-count*
- *request chassis cluster failover node*
- *request chassis cluster failover reset*

**List of Sample Output**
show chassis cluster status on page 406
show chassis cluster status with preemptive delay on page 407
show chassis cluster status redundancy-group 1 on page 407

**Output Fields** Table 40 on page 405 lists the output fields for the **show chassis cluster status** command. Output fields are listed in the approximate order in which they appear.

### Table 40: show chassis cluster status Output Fields

| Field Name | Field Description |
|---|---|
| **Cluster ID** | ID number (1-15) of a cluster is applicable for releases upto Junos OS Release 12.1X45-D10. ID number (1-255) is applicable for Releases 12.1X45-D10 and later. Setting a cluster ID to 0 is equivalent to disabling a cluster. |
| **Redundancy-Group** | You can create up to 128 redundancy groups in the chassis cluster. |
| **Node name** | Node (device) in the chassis cluster (**node0** or **node1**). |
| **Priority** | Assigned priority for the redundancy group on that node. |

Table 40: show chassis cluster status Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| **Status** | State of the redundancy group (**Primary**, **Secondary**, **Lost**, or **Unavailable**).<br>• **Primary**—Redundancy group is active and passing traffic.<br>• **Secondary**—Redundancy group is passive and not passing traffic.<br>• **Lost**—Node loses contact with the other node through the control link. Most likely to occur when both nodes are in a cluster and there is a control link failure, one node cannot exchange heartbeats, or when the other node is rebooted.<br>• **Unavailable**—Node has not received a single heartbeat over the control link from the other node since the other node booted up. Most likely to occur when one node boots up before the other node, or if only one node is present in the cluster. |
| **Preempt** | • **Yes**: Primary state can be preempted based on priority.<br>• **No**: Change in priority will not preempt the primary state. |
| **Manual failover** | • **Yes**: Primary state is set manually through the CLI with the **request chassis cluster failover node** or **request chassis cluster failover redundancy-group** command. This overrides **Priority** and **Preempt**.<br>• **No**: Primary state is not set manually through the CLI. |
| Monitor-failures | • **None**: Cluster working properly.<br>• **Monitor Failure code**: Cluster is not working properly and the respective failure code is displayed. |

## Sample Output

### show chassis cluster status

```
user@host> show chassis cluster status

Monitor Failure codes:
  CS  Cold Sync monitoring        FL  Fabric Connection monitoring
    GR  GRES monitoring             HW  Hardware monitoring
    IF  Interface monitoring        IP  IP monitoring
    LB  Loopback monitoring         MB  Mbuf monitoring
    NH  Nexthop monitoring          NP  NPC monitoring
    SP  SPU monitoring              SM  Schedule monitoring
    CF  Config Sync monitoring

Cluster ID: 1
Node   Priority Status         Preempt Manual   Monitor-failures

Redundancy group: 0 , Failover count: 1
node0  200      primary        no      no       None
node1  1        secondary      no      no       None

Redundancy group: 1 , Failover count: 1
node0  101      primary        no      no       None
node1  1        secondary      no      no       None
```

## Sample Output

### show chassis cluster status with preemptive delay

```
user@host> show chassis cluster status

Cluster ID: 1
Node   Priority Status         Preempt Manual   Monitor-failures
Redundancy group: 0, Failover count: 1
node0  200      primary        no      no       None
node1  100      secondary      no      no       None
Redundancy group: 1, Failover count: 3
node0 200    primary-preempt-hold    yes   no    None node1   100       secondary
                yes      no       None
```

## Sample Output

### show chassis cluster status redundancy-group 1

```
user@host> show chassis cluster status redundancy-group 1

Monitor Failure codes:
    CS  Cold Sync monitoring        FL  Fabric Connection monitoring
    GR  GRES monitoring             HW  Hardware monitoring
    IF  Interface monitoring        IP  IP monitoring
    LB  Loopback monitoring         MB  Mbuf monitoring
    NH  Nexthop monitoring          NP  NPC monitoring
    SP  SPU monitoring              SM  Schedule monitoring
    CF  Config Sync monitoring

Cluster ID: 1
Node   Priority Status         Preempt Manual   Monitor-failures

Redundancy group: 1 , Failover count: 1
node0  101      primary        no      no       None
node1  1        secondary      no      no       None
```

## show dot1x authentication-bypassed-users

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show dot1x authentication-bypassed-users

**Release Information** Command introduced in Junos OS Release 15.1X49-D80.

**Description** Display the supplicants (users) that have bypassed 802.1X authentication.

**Required Privilege Level** view

**Related Documentation**
- show dot1x authentication-failed-users on page 409
- dot1x on page 321

**List of Sample Output** show dot1x authentication-bypassed-users on page 408

**Output Fields** Table 41 on page 408 lists the output fields for the show dot1x authentication-bypassed-users command. Output fields are listed in the approximate order in which they appear.

Table 41: show dot1x authentication-bypassed-users Output Fields

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Interface** | The MAC address configured to bypass 802.1X authentication. | **all** |
| **MAC address** | The MAC address configured statically on the interface. | **all** |
| **VLAN** | The VLAN that is configured to bypass 802.1X authentication. | **all** |

## Sample Output

### show dot1x authentication-bypassed-users

```
user@host> show dot1x authentication-bypassed-users

MAC address           Interface      VLAN
00:50:56:85:66:0f     ge-0/0/0.0     vlan6
00:50:56:9e:56:42     ge-0/0/1.0     vlan6
```

## show dot1x authentication-failed-users

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show dot1x authentication-failed-users

**Release Information** Command introduced in Junos OS Release 15.1X49-D80.

**Description** Display the supplicants (users) that have failed 802.1X authentication.

**Required Privilege Level** view

**Related Documentation**
- show dot1x authentication-bypassed-users on page 408
- dot1x on page 321

**List of Sample Output** show dot1x authentication-failed-users on page 409

**Output Fields** Table 42 on page 409 lists the output fields for the **show dot1x authentication-failed-users** command. Output fields are listed in the approximate order in which they appear.

Table 42: show dot1x authentication-failed-users Output Fields

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Interface** | The MAC address configured to bypass 802.1X authentication. | **all** |
| **MAC address** | The MAC address configured statically on the interface. | **all** |
| **User** | The user that is configured on the RADIUS server and that has failed 802.1X authentication. | **all** |
| **Failure Count** | The number of times that 802.1X authentication has failed on the interface. | **all** |

### Sample Output

#### show dot1x authentication-failed-users

```
user@host> show dot1x authentication-failed-users

Interface     MAC address          User            Failure Count
ge-0/0/0.0    00:50:56:85:66:0f    00505685660f    1
ge-0/0/1.0    00:50:56:9e:56:42    0050569e5642    1
```

## show dot1x interface

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** **show dot1x interface** <<*interface-name*>
<brief | detail>

**Release Information** Command introduced in Junos OS Release 15.1X49-D80.

**Description** Display the current operational state of all ports with the list of connected users.

This command displays the list of connected supplicants received from the RADIUS authentication server regardless of the session state—that is, for both authenticated supplicants and for supplicants that attempted authentication.

**Options** **none**—Display information for all authenticator ports.

**brief | detail**—(Optional) Display the specified level of output.

**interface *interface-name***—(Optional) Display information for the specified interface with a list of connected supplicants.

**Required Privilege Level** view

**Related Documentation**
- show dot1x authentication-bypassed-users on page 408
- dot1x on page 321

**List of Sample Output**
show dot1x interface brief on page 414
show dot1x interface detail on page 414

**Output Fields** Table 43 on page 410 lists the output fields for the **show dot1x interface** command. Output fields are listed in the approximate order in which they appear.

### Table 43: show dot1x interface Output Fields

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Interface** | Name of a port. | **all** |
| **MAC address** | The MAC address of the connected supplicant on the port. | **all** |
| **Role** | The 802.1X authentication role of the interface. When 802.1X is enabled on an interface, the role is **Authenticator**. As **Authenticator**, the interface blocks LAN access until a supplicant is authenticated through 802.1X or MAC RADIUS authentication. | **brief**, **detail** |

Table 43: show dot1x interface Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **State** | The state of the port:<br>• **Authenticated**—The supplicant has been authenticated through the RADIUS server or has been permitted access through server fail fallback.<br>• **Authenticating**—The supplicant is authenticating through the RADIUS server.<br>• **Held**—An action has been triggered through server fail fallback during a RADIUS server timeout. A supplicant is denied access, permitted access through a specified VLAN, or maintains the authenticated state granted to it before the RADIUS server timeout occurred. | **brief** |
| **User** | The username of the connected supplicant. | **brief** |
| **Administrative state** | The administrative state of the port:<br>• **auto**—Traffic is allowed through the port based on the authentication result (by default).<br>• **force-authorize**—All traffic flows through the port irrespective of the authentication result. This state is not allowed on an interface whose VLAN membership has been set to **dynamic**.<br>• **force-unauthorize**—All traffic drops on the port irrespective of the authentication result. This state is not allowed on an interface whose VLAN membership has been set to **dynamic**. | **detail** |
| **Supplicant** | The mode for the supplicant:<br>• **single**—Only the first supplicant is authenticated. All other supplicants that connect later to the port are allowed full access without any further authentication. They effectively *piggyback* on the first supplicant's authentication.<br>• **single-secure**—Only one supplicant is allowed to connect to the port. No other supplicant is allowed to connect until the first supplicant logs out.<br>• **multiple**—Multiple supplicants are allowed to connect to the port. Each supplicant is authenticated individually. | **detail** |
| **Quiet period** | The number of seconds the port waits before reattempting authentication after a failed authentication exchange with the supplicant. | **detail** |
| **Transmit period** | The number of seconds the port waits before retransmitting the initial EAPOL PDUs to the supplicant. | **detail** |
| **MAC Radius** | MAC RADIUS authentication:<br>• **enabled**—The device sends an EAPOL request to the connecting host to attempt 802.1X authentication and if the connecting host is unresponsive, the device tries to authenticate the host by using the MAC address.<br>• **disabled**—The default. The device does not attempt to authenticate the MAC address of the connecting host. | **detail** |

Table 43: show dot1x interface Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **MAC Radius authentication protocol** | MAC RADIUS authentication protocol:<br>• **EAP-MD5**—The EAP-MD5 protocol is used for MAC RADIUS authentication. EAP-MD5 is an authentication method belonging to the Extensible Authentication Protocol (EAP) authentication framework. EAP-MD5 is the default authentication protocol.<br>• **PAP**—The Password Authentication Protocol (PAP) authentication protocol is used for MAC RADIUS authentication. | **detail** |
| **MAC Radius restrict** | The authentication method is restricted to MAC RADIUS. 802.1X authentication is not enabled. | **detail** |
| **Reauthentication** | The reauthentication state:<br>• **disable**—Periodic reauthentication of the client is disabled.<br>• **interval**—Sets the periodic reauthentication time interval. | **detail** |
| **Supplicant timeout** | The number of seconds the port waits for a response when relaying a request from the authentication server to the supplicant before resending the request. | **detail** |
| **Server timeout** | The number of seconds the port waits for a reply when relaying a response from the supplicant to the authentication server before timing out. | **detail** |
| **Maximum EAPOL requests** | The maximum number of times an EAPOL request packet is retransmitted to the supplicant before the authentication session times out. | **detail** |
| **Number of clients bypassed because of authentication** | The number of non-802.1X clients granted access to the LAN by means of static MAC bypass. The following fields are displayed:<br>• **Client**—MAC address of the client.<br>• **vlan**—The name of the VLAN to which the client is connected. | **detail** |
| **Guest VLAN member** | The VLAN to which a supplicant is connected when the supplicant is authenticated using a guest VLAN. If a guest VLAN is not configured on the interface, this field displays <**not configured**>. | **detail** |
| **Number of connected supplicants** | The number of supplicants connected to a port. | **detail** |
| **Supplicant** | The username and MAC address of the connected supplicant. | **detail** |

Table 43: show dot1x interface Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Authentication method** | The authentication method used for a supplicant:<br>• **CWA Authentication**—A supplicant is authenticated by the central Web authentication (CWA) server.<br>• **Guest VLAN**—A supplicant is connected to the LAN through the guest VLAN.<br>• **MAC RADIUS**—A nonresponsive host is authenticated based on its MAC address. The MAC address is configured as permitted on the RADIUS server. The RADIUS server lets the device know that the MAC address is a permitted address, and the device opens LAN access to the nonresponsive host on the interface to which it is connected.<br>• **RADIUS**—A supplicant is configured on the RADIUS server, the RADIUS server communicates this to the device, and the device opens LAN access on the interface to which the supplicant is connected.<br>• **Server-fail**—One of the following fallback actions is in effect because the RADIUS server is unreachable. Indicates whether EAPOL block is in effect, and the amount of time remaining for EAPOL block (in seconds).<br>  • **deny**—The supplicant is denied access to the LAN, preventing traffic from flowing from the supplicant through the interface. This is the default server fail fallback action.<br>  • **permit**—The supplicant is permitted access to the LAN as if the supplicant had been successfully authenticated by the RADIUS server.<br>  • **use-cache**—In the event that the RADIUS server times out when the supplicant is attempting reauthentication, the supplicant is reauthenticated only if it was previously authenticated; otherwise, the supplicant is denied LAN access.<br>  • **VLAN**—The supplicant is configured to be moved to a specified VLAN if the RADIUS server is unavailable to reauthenticate the supplicant. (The VLAN must already exist on the device.) | **detail** |
| **Authenticated VLAN** | The VLAN to which the supplicant is connected. | **detail** |
| **Dynamic filter** | User policy filter sent by the RADIUS server. | **detail** |
| **Session Reauth interval** | The configured reauthentication interval. | **detail** |
| **Reauthentication due in** | The number of seconds in which reauthentication occurs again for the connected supplicant. | **detail** |
| **Session Accounting Interim Interval** | The number of seconds between interim RADIUS accounting messages. | **detail** |
| **Accounting Update due in** | The number of seconds until the next interim RADIUS accounting update is due. | **detail** |
| **CWA Redirect URL** | The URL used to redirect the supplicant to a central Web server for authentication. | **detail** |

## Sample Output

### show dot1x interface brief

```
user@root> show dot1x interface brief
802.1X Information:
Interface     Role          State          MAC address          User
ge-0/0/1      Authenticator Connecting     00:50:56:85:66:0F    00505685660f
ge-0/0/2      Authenticator Authenticated  00:50:56:9E:56:42    0050569e5642
```

### show dot1x interface detail

```
user@root> show dot1x interface detail

 ge-0/0/0.0
  Role: Authenticator
  Administrative state: Auto
  Supplicant mode: Multiple
  Number of retries: 3
  Quiet period: 60 seconds
  Transmit period: 30 seconds
  Mac Radius: Enabled
  Mac Radius Restrict: Disabled
  Reauthentication: Enabled
  Configured Reauthentication interval: 30 seconds
  Supplicant timeout: 30 seconds
  Server timeout: 30 seconds
  Maximum EAPOL requests: 2
  Guest VLAN member: not configured
  Number of connected supplicants: 1
    Supplicant: 00505685660f, 00:50:56:85:66:0F
      Operational state: Authenticated
      Backend Authentication state: Idle
      Authentication method: Server-Reject Vlan
      Authenticated VLAN: visitor-vlan
      Session Reauth interval: 30 seconds
      Reauthentication due in 20 seconds
ge-0/0/1.0
  Role: Authenticator
  Administrative state: Auto
  Supplicant mode: Multiple
  Number of retries: 3
  Quiet period: 60 seconds
  Transmit period: 30 seconds
  Mac Radius: Enabled
  Mac Radius Restrict: Disabled
  Reauthentication: Enabled
  Configured Reauthentication interval: 30 seconds
  Supplicant timeout: 30 seconds
  Server timeout: 30 seconds
  Maximum EAPOL requests: 2
  Guest VLAN member: not configured
  Number of connected supplicants: 1
    Supplicant: 0050569e5642, 00:50:56:9E:56:42
      Operational state: Authenticated
      Backend Authentication state: Idle
      Authentication method: Server-Reject Vlan
      Authenticated VLAN: visitor-vlan
      Session Reauth interval: 30 seconds
      Reauthentication due in 24 seconds
```

## show dot1x static-mac-address

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show dot1x static-mac-address <interface *interface-name*>

**Release Information** Command introduced in Junos OS Release 15.1X49-D80.

**Description** Display all the static MAC addresses of interfaces that are configured to bypass 802.1X authentication.

**Options** **none**—Display static MAC addresses for all interfaces.

**interface *interface-name***—(Optional) Display static MAC addresses for a specific interface.

**Required Privilege Level** view

**Related Documentation**
- show dot1x authentication-bypassed-users on page 408
- dot1x on page 321

**List of Sample Output**
show dot1x static-mac-address on page 416
show dot1x static-mac-address interface (Specific Interface) on page 417

**Output Fields** Table 44 on page 416 lists the output fields for the **show dot1x static-mac-address** command. Output fields are listed in the approximate order in which they appear.

### Table 44: show dot1x static-mac-address Output Fields

| Field Name | Field Description | Level of Output |
|---|---|---|
| **MAC address/prefix** | The MAC address of the device that is configured to bypass 802.1X authentication. | **all** |
| **VLAN-Assignment** | The name of the VLAN to which the device is assigned. | **all** |
| **Interface** | The name of the interface on which authentication is bypassed for a given MAC address. | **all** |

## Sample Output

### show dot1x static-mac-address

```
user@host> show dot1x static-mac-address

MAC address/prefix          VLAN-Assignment          Interface
00:50:56:85:66:0f/48        vlan6                    ge-0/0/0.0
00:50:56:9e:56:42/48        vlan6                    ge-0/0/1.0
```

### show dot1x static-mac-address interface (Specific Interface)

```
user@host> show dot1x static-mac-address interface ge-0/0/0

MAC address/prefix              VLAN-Assignment           Interface
00:50:56:85:66:0f/48            vlan6                     ge-0/0/0.0
```

## show dot1x statistics

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** **show dot1x statistics interface <*interface-name*>**

**Release Information** Command introduced in Junos OS Release 15.1X49-D80.

**Description** Display 802.1X statistics on this interface.

**Options** **interface *interface-name***—(Optional) Displays statistical information for the interface.

**Required Privilege Level** view

**Related Documentation**
- dot1x on page 321
- show dot1x authentication-bypassed-users on page 408

**List of Sample Output** show dot1x statistics interface on page 418

### Sample Output

#### show dot1x statistics interface

```
user@host> show dot1x statistics interface ge-0/0/0

Interface: ge-0/0/0.0
  TxReqId = 4 TxReq = 0 TxTotal = 4
  RxStart = 0 RxLogoff = 0 RxRespId = 0 RxResp = 0
  RxInvalid = 0 RxLenErr = 0 RxTotal = 0
  LastRxVersion = 0 LastRxSrcMac = 00:50:56:85:66:0f
```

## show ethernet-switching mac-learning-log (View)

**Supported Platforms** SRX Series

**Syntax** **show ethernet-switching mac-learning-log**

**Release Information** Command introduced in Junos OS Release 9.5.

**Description** Displays the event log of learned MAC addresses.

**Required Privilege Level** view

**Related Documentation**
- show ethernet-switching table (View) on page 421

**Output Fields** Table 45 on page 419 lists the output fields for the show ethernet-switching mac-learning-log command. Output fields are listed in the approximate order in which they appear.

### Table 45: show ethernet-switching-mac-learning-log Output Fields

| Field Name | Field Description |
|---|---|
| **Date and Time** | Timestamp when the MAC address was added or deleted from the log. |
| **VLAN-IDX** | VLAN index. An internal value assigned by Junos OS for each VLAN. |
| **MAC** | Learned MAC address. |
| **Deleted \| Added** | MAC address deleted or added to the MAC learning log. |
| **Blocking** | The forwarding state of the interface:<br>• blocked—Traffic is not being forwarded on the interface.<br>• unblocked—Traffic is forwarded on the interface. |

## Sample Output

### show ethernet-switching mac-learning-log

```
user@host> show ethernet-switching mac-learning-log
Wed Mar 18 08:07:05 2009
vlan_idx 7 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 9 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 10 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 11 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
```

```
vlan_idx 12 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 13 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 14 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 15 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 16 mac 00:00:5E:00:53:00 was deleted
Wed Mar 18 08:07:05 2009
vlan_idx 4 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 6 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 7 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 9 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 10 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 11 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 12 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 13 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 14 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 15 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 16 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 5 mac 00:00:5E:00:53:00 was added
Wed Mar 18 08:07:05 2009
vlan_idx 18 mac 00:00:5E:00:53:AA was learned
Wed Mar 18 08:07:05 2009
vlan_idx 5 mac 00:00:5E:00:53:AB was learned
Wed Mar 18 08:07:05 2009
vlan_idx 6 mac 00:00:5E:00:53:AC was learned
Wed Mar 18 08:07:05 2009
vlan_idx 16 mac 00:00:5E:00:53:AD was learned
Wed Mar 18 08:07:05 2009
vlan_idx 7 mac 00:00:5E:00:53:AE was learned
Wed Mar 18 08:07:05 2009
vlan_idx 8 mac 00:00:5E:00:53:AF was learned
Wed Mar 18 08:07:05 2009
vlan_idx 12 mac 00:00:5E:00:53:AG was learned
[output truncated]
```

## show ethernet-switching table (View)

**Supported Platforms** SRX Series

**Syntax** **show ethernet-switching table (brief |detail |extensive) interface** *interface-name*

**Release Information** Command introduced in Junos OS Release 9.5.

**Description** Displays the Ethernet switching table.

**Options**

- **none**—(Optional) Display brief information about the Ethernet switching table.
- **brief | detail | extensive**—(Optional) Display the specified level of output.
- **interface-name**—(Optional) Display the Ethernet switching table for a specific interface.

**Required Privilege Level** view

**Related Documentation**

- show ethernet-switching mac-learning-log (View) on page 419

**Output Fields** Table 46 on page 421 lists the output fields for the **show ethernet-switching table** command. Output fields are listed in the approximate order in which they appear.

### Table 46: show ethernet-switching table Output Fields

| Field Name | Field Description |
|---|---|
| **VLAN** | The name of a VLAN. |
| **MAC address** | The MAC address associated with the VLAN. |
| **Type** | The type of MAC address. Values are:<br>• static—The MAC address is manually created.<br>• learn—The MAC address is learned dynamically from a packet's source MAC address.<br>• flood—The MAC address is unknown and flooded to all members. |
| **Age** | The time remaining before the entry ages out and is removed from the Ethernet switching table. |
| **Interfaces** | Interface associated with learned MAC addresses or All-members (flood entry). |
| **Learned** | For learned entries, the time which the entry was added to the Ethernet switching table. |

## Sample Output

### show ethernet-switching table

```
user@host> show ethernet-switching table
Ethernet-switching table: 57 entries, 17 learned
VLAN MAC address Type Age Interfaces
F2 * Flood - All-members
F2 00:00:5E:00:53:AC Learn 0 ge-0/0/44.0
F2 00:00:5E:00:53:AD Static - Router
Linux * Flood - All-members
Linux 00:00:5E:00:53:AE Static - Router
Linux 00:00:5E:00:53:AF Learn 0 ge-0/0/47.0
T1 * Flood - All-members
T1 00:00:5E:00:53:AA Learn 0 ge-0/0/46.0
T1 00:00:5E:00:53:AB Static - Router
T1 00:00:5E:00:53:AC Learn 0 ge-0/0/46.0
T1 00:00:5E:00:53:AD Static - Router
T10 * Flood - All-members
T10 00:00:5E:00:53:AE Static - Router
T10 00:00:5E:00:53:AF Learn 0 ge-0/0/46.0
T10 00:00:5E:00:53:AG Static - Router
T111 * Flood - All-members
T111 00:00:5E:00:53:AH Learn 0 ge-0/0/15.0
T111 00:00:5E:00:53:AI Static - Router
T111 00:00:5E:00:53:AJ Learn 0 ge-0/0/15.0
T2 * Flood - All-members
T2 00:00:5E:00:53:AK Static - Router
T2 00:00:5E:00:53:AL Learn 0 ge-0/0/46.0
T2 00:00:5E:00:53:AM Static - Router
T3 * Flood - All-members
T3 00:00:5E:00:53:AN Static - Router
T3 00:00:5E:00:53:AO Learn 0 ge-0/0/46.0
T3 00:00:5E:00:53:AP Static - Router
T4 * Flood - All-members
T4 00:00:5E:00:53:AQ Static - Router
T4 00:00:5E:00:53:AR Learn 0 ge-0/0/46.0
[output truncated]
```

## Sample Output

### show ethernet-switching table brief

```
user@host> show ethernet-switching table brief
Ethernet-switching table: 57 entries, 17 learned
VLAN MAC address Type Age Interfaces
F2 * Flood - All-members
F2 00:00:5E:00:53:AC Learn 0 ge-0/0/44.0
F2 00:00:5E:00:53:AE Static - Router
Linux * Flood - All-members
Linux 00:00:5E:00:53:AA Static - Router
Linux 00:00:5E:00:53:AB Learn 0 ge-0/0/47.0
T1 * Flood - All-members
T1 00:00:5E:00:53:AC Learn 0 ge-0/0/46.0
T1 00:00:5E:00:53:AD Static - Router
T1 00:00:5E:00:53:AE Learn 0 ge-0/0/46.0
T1 00:00:5E:00:53:AF Static - Router
T10 * Flood - All-members
T10 00:00:5E:00:53:AG Static - Router
T10 00:00:5E:00:53:AH Learn 0 ge-0/0/46.0
```

```
T10 00:00:5E:00:53:AI Static - Router
T111 * Flood - All-members
T111 00:00:5E:00:53:AJ Learn 0 ge-0/0/15.0
T111 00:00:5E:00:53:AK Static - Router
T111 00:00:5E:00:53:AL Learn 0 ge-0/0/15.0
T2 * Flood - All-members
T2 00:00:5E:00:53:AM Static - Router
T2 00:00:5E:00:53:AN Learn 0 ge-0/0/46.0
T2 00:00:5E:00:53:AO Static - Router
T3 * Flood - All-members
T3 00:00:5E:00:53:AP Static - Router
T3 00:00:5E:00:53:AQ Learn 0 ge-0/0/46.0
T3 00:00:5E:00:53:AR Static - Router
T4 * Flood - All-members
T4 00:00:5E:00:53:AS Static - Router
T4 00:00:5E:00:53:AT Learn 0 ge-0/0/46.0
[output truncated]
```

## Sample Output

### show ethernet-switching table detail

```
user@host> show ethernet-switching table detail
Ethernet-switching table: 57 entries, 17 learned
F2, *
Interface(s): ge-0/0/44.0
Type: Flood
F2, 00:00:5E:00:53:AC
Interface(s): ge-0/0/44.0
Type: Learn, Age: 0, Learned: 2:03:09
F2, 00:00:5E:00:53:AA
Interface(s): Router
Type: Static
Linux, *
Interface(s): ge-0/0/47.0
Type: Flood
Linux, 00:00:5E:00:53:AB
Interface(s): Router
Type: Static
Linux, 00:00:5E:00:53:AC
Interface(s): ge-0/0/47.0
Type: Learn, Age: 0, Learned: 2:03:08
T1, *
Interface(s): ge-0/0/46.0
Type: Flood
T1, 00:00:5E:00:53:AD
Interface(s): ge-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:07
T1, 00:00:5E:00:53:AE
Interface(s): Router
Type: Static
T1, 00:00:5E:00:53:AF
Interface(s): ge-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:07
T1, 00:00:5E:00:53:AG
Interface(s): Router
Type: Static
T10, *
Interface(s): ge-0/0/46.0
Type: Flood
T10, 00:00:5E:00:53:AH
```

```
Interface(s): Router
Type: Static
T10, 00:00:5E:00:53:AI
Interface(s): ge-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:08
T10, 00:00:5E:00:53:AJ
Interface(s): Router
Type: Static
T111, *
Interface(s): ge-0/0/15.0
Type: Flood
[output truncated]
```

## Sample Output

### show ethernet-switching table extensive

```
user@host> show ethernet-switching table extensive
Ethernet-switching table: 57 entries, 17 learned
F2, *
Interface(s): ge-0/0/44.0
Type: Flood
F2, 00:00:5E:00:53:AC
Interface(s): ge-0/0/44.0
Type: Learn, Age: 0, Learned: 2:03:09
F2, 00:00:5E:00:53:AA
Interface(s): Router
Type: Static
Linux, *
Interface(s): ge-0/0/47.0
Type: Flood
Linux, 00:00:5E:00:53:AB
Interface(s): Router
Type: Static
Linux, 00:00:5E:00:53:AC
Interface(s): ge-0/0/47.0
Type: Learn, Age: 0, Learned: 2:03:08
T1, *
Interface(s): ge-0/0/46.0
Type: Flood
T1, 00:00:5E:00:53:AD
Interface(s): ge-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:07
T1, 00:00:5E:00:53:AE
Interface(s): Router
Type: Static
T1, 00:00:5E:00:53:AF
Interface(s): ge-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:07
T1, 00:00:5E:00:53:AG
Interface(s): Router
Type: Static
T10, *
Interface(s): ge-0/0/46.0
Type: Flood
T10, 00:00:5E:00:53:AH
Interface(s): Router
Type: Static
T10, 00:00:5E:00:53:AI
Interface(s): ge-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:08
```

```
T10, 00:00:5E:00:53:AJ
Interface(s): Router
Type: Static
T111, *
Interface(s): ge-0/0/15.0
Type: Flood
[output truncated]
```

## Sample Output

### show ethernet-switching table interface ge-0/0/1

```
user@host> show ethernet-switching table interface ge-0/0/1
Ethernet-switching table: 1 unicast entries
VLAN     MAC address         Type     Age Interfaces
V1       *                   Flood    - All-members
V1       00:00:5E:00:53:AF   Learn    0 ge-0/0/1.0
```

## show interfaces (SRX Series)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
show interfaces (
  <interface-name>
  <brief | detail | extensive | terse>
  <controller interface-name>|
  <descriptions interface-name>|
  <destination-class (all | destination-class-name logical-interface-name)>|
  <diagnostics optics interface-name>|
  <far-end-interval interface-fpc/pic/port>|
  <filters interface-name>|
  <flow-statistics interface-name>|
  <interval interface-name>|
  <load-balancing (detail | interface-name)>|
  <mac-database mac-address mac-address>|
  <mc-ae id identifier unit number revertive-info>|
  <media interface-name>|
  <policers interface-name>|
  <queue both-ingress-egress egress forwarding-class forwarding-class ingress l2-statistics>|
  <redundancy (detail | interface-name)>|
  <routing brief detail summary interface-name>|
  <routing-instance (all | instance-name)>|
  <snmp-index snmp-index>|
  <source-class (all | destination-class-name logical-interface-name)>|
  <statistics interface-name>|
  <switch-port switch-port number>|
  <transport pm (all | optics | otn) (all | current | currentday | interval | previousday) (all |
    interface-name)>|
  <zone interface-name>
)
```

**Release Information** Command modified in Junos OS Release 9.5.

**Description** Display status information and statistics about interfaces on SRX Series appliance running Junos OS.

On SRX Series appliance, on configuring identical IPs on a single interface, you will not see a warning message; instead, you will see a syslog message.

**Options**

- **interface-name**—(Optional) Display standard information about the specified interface. Following is a list of typical interface names. Replace pim with the PIM slot and port with the port number.
  - **at-*pim*/0/*port***—ATM-over-ADSL or ATM-over-SHDSL interface.
  - **ce1-*pim*/0/ *port***—Channelized E1 interface.
  - **cl-0/0/8**—3G wireless modem interface for SRX320 devices.
  - **ct1-*pim*/0/*port***—Channelized T1 interface.

  - **dl0**—Dialer Interface for initiating ISDN and USB modem connections.
  - **e1-*pim*/0/*port***—E1 interface.
  - **e3-*pim*/0/*port***—E3 interface.
  - **fe-*pim*/0/*port***—Fast Ethernet interface.
  - **ge-*pim*/0/*port***—Gigabit Ethernet interface.
  - **se-*pim*/0/*port***—Serial interface.
  - **t1-*pim*/0/*port***—T1 (also called DS1) interface.
  - **t3-*pim*/0/*port***—T3 (also called DS3) interface.
  - **wx-*slot*/0/0**—WAN acceleration interface, for the WXC Integrated Services Module (ISM 200).

- **brief | detail | extensive | terse**—(Optional) Display the specified level of output.
- **controller**—(Optional) Show controller information.
- **descriptions**—(Optional) Display interface description strings.
- **destination-class**—(Optional) Show statistics for destination class.
- **diagnostics**—(Optional) Show interface diagnostics information.
- **far-end-interval**—(Optional) Show far end interval statistics.
- **filters**—(Optional) Show interface filters information.
- **flow-statistics**—(Optional) Show security flow counters and errors.
- **interval**—(Optional) Show interval statistics.
- **load-balancing**—(Optional) Show load-balancing status.
- **mac-database**—(Optional) Show media access control database information.
- **mc-ae**—(Optional) Show MC-AE configured interface information.
- **media**—(Optional) Display media information.
- **policers**—(Optional) Show interface policers information.
- **queue**—(Optional) Show queue statistics for this interface.
- **redundancy**—(Optional) Show redundancy status.
- **routing**—(Optional) Show routing status.
- **routing-instance**—(Optional) Name of routing instance.
- **snmp-index**—(Optional) SNMP index of interface.
- **source-class**—(Optional) Show statistics for source class.
- **statistics**—(Optional) Display statistics and detailed output.
- **switch-port**—(Optional) Front end port number (0..15).

- **transport**—(Optional) Show interface transport information.
- **zone**—(Optional) Interface's zone.

**Required Privilege Level** view

**Related Documentation**
- Understanding Layer 2 Interfaces on Security Devices on page 5

**List of Sample Output**

show security zones on page 457

**Output Fields** Table 47 on page 429 lists the output fields for the **show interfaces** command. Output fields are listed in the approximate order in which they appear.

## Table 47: show interfaces Output Fields

| Field Name | Field Description | Level of Output |
|---|---|---|
| Physical Interface | | |
| **Physical interface** | Name of the physical interface. | All levels |
| **Enabled** | State of the interface. | All levels |
| **Interface index** | Index number of the physical interface, which reflects its initialization sequence. | **detail extensive** none |
| **SNMP ifIndex** | SNMP index number for the physical interface. | **detail extensive** none |
| **Link-level type** | Encapsulation being used on the physical interface. | All levels |
| **Generation** | Unique number for use by Juniper Networks technical support only. | **detail extensive** |
| **MTU** | Maximum transmission unit size on the physical interface. | All levels |
| **Link mode** | Link mode: Full-duplex or Half-duplex. | |
| **Speed** | Speed at which the interface is running. | All levels |
| **BPDU error** | Bridge protocol data unit (BPDU) error: Detected or None | |
| **Loopback** | Loopback status: **Enabled** or **Disabled**. If loopback is enabled, type of loopback: **Local** or **Remote**. | All levels |
| **Source filtering** | Source filtering status: **Enabled** or **Disabled**. | All levels |
| **Flow control** | Flow control status: **Enabled** or **Disabled**. | All levels |
| **Auto-negotiation** | (Gigabit Ethernet interfaces) Autonegotiation status: **Enabled** or **Disabled**. | All levels |
| **Remote-fault** | (Gigabit Ethernet interfaces) Remote fault status:<br>• **Online**—Autonegotiation is manually configured as online.<br>• **Offline**—Autonegotiation is manually configured as offline. | All levels |
| **Device flags** | Information about the physical device. | All levels |
| **Interface flags** | Information about the interface. | All levels |
| **Link flags** | Information about the physical link. | All levels |
| **CoS queues** | Number of CoS queues configured. | **detail extensive** none |

Table 47: show interfaces Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Current address** | Configured MAC address. | **detail extensive** none |
| **Last flapped** | Date, time, and how long ago the interface went from down to up. The format is **Last flapped:** ***year-month-day hour:minute:second:timezone*** **(*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).** | **detail extensive** none |
| **Input Rate** | Input rate in bits per second (bps) and packets per second (pps). | None |
| **Output Rate** | Output rate in bps and pps. | None |
| **Active alarms and Active defects** | Ethernet-specific defects that can prevent the interface from passing packets. When a defect persists for a certain amount of time, it is promoted to an alarm. These fields can contain the value **None** or **Link**.<br>• **None**—There are no active defects or alarms.<br>• **Link**—Interface has lost its link state, which usually means that the cable is unplugged, the far-end system has been turned off, or the PIC is malfunctioning. | **detail extensive** none |
| **Statistics last cleared** | Time when the statistics for the interface were last set to zero. | **detail extensive** |
| **Traffic statistics** | Number and rate of bytes and packets received and transmitted on the physical interface.<br>• **Input bytes**—Number of bytes received on the interface.<br>• **Output bytes**—Number of bytes transmitted on the interface.<br>• **Input packets**—Number of packets received on the interface.<br>• **Output packets**—Number of packets transmitted on the interface. | **detail extensive** |

Table 47: show interfaces Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Input errors** | Input errors on the interface.<br>• **Errors**—Sum of the incoming frame aborts and FCS errors.<br>• **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism.<br>• **Framing errors**—Number of packets received with an invalid frame checksum (FCS).<br>• **Runts**—Number of frames received that are smaller than the runt threshold.<br>• **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that Junos OS does not handle.<br>• **L3 incompletes**—Number of incoming packets discarded because they failed Layer 3 (usually IPv4) sanity checks of the header. For example, a frame with less than 20 bytes of available IP header is discarded. L3 incomplete errors can be ignored by configuring the **ignore-l3-incompletes** .<br>• **L2 channel errors**—Number of times the software did not find a valid logical interface for an incoming frame.<br>• **L2 mismatch timeouts**—Number of malformed or short packets that caused the incoming packet handler to discard the frame as unreadable.<br>• **FIFO errors**—Number of FIFO errors in the receive direction that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning.<br>• **Resource errors**—Sum of transmit drops. | **extensive** |
| **Output errors** | Output errors on the interface.<br>• **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC or PIM is malfunctioning.<br>• **Errors**—Sum of the outgoing frame aborts and FCS errors.<br>• **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism.<br>• **Collisions**—Number of Ethernet collisions. The Gigabit Ethernet PIC supports only full-duplex operation; therefore, for Gigabit Ethernet PICs, this number must always remain 0. If it is nonzero, there is a software bug.<br>• **Aged packets**—Number of packets that remained in shared packet SDRAM so long that the system automatically purged them. The value in this field must never increment. If it does, it is most likely a software bug or possibly malfunctioning hardware.<br>• **FIFO errors**—Number of FIFO errors in the send direction as reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning.<br>• **HS link CRC errors**—Number of errors on the high-speed links between the ASICs responsible for handling the interfaces.<br>• **MTU errors**—Number of packets whose size exceeded the MTU of the interface.<br>• **Resource errors**—Sum of transmit drops. | **extensive** |

Table 47: show interfaces Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Ingress queues** | Total number of ingress queues supported on the specified interface. | **extensive** |
| **Queue counters and queue number** | CoS queue number and its associated user-configured forwarding class name.<br>• **Queued packets**—Number of queued packets.<br>• **Transmitted packets**—Number of transmitted packets.<br>• **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism. | **detail extensive** |
| **MAC statistics** | **Receive** and **Transmit** statistics reported by the PIC's MAC subsystem, including the following:<br>• **Total octets** and **total packets**—Total number of octets and packets.<br>• **Unicast packets, Broadcast packets,** and **Multicast packets**—Number of unicast, broadcast, and multicast packets.<br>• **CRC/Align errors**—Total number of packets received that had a length (excluding framing bits, but including FCS octets) of between 64 and 1518 octets, inclusive, and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error).<br>• **FIFO error**—Number of FIFO errors that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC or a cable is probably malfunctioning.<br>• **MAC control frames**—Number of MAC control frames.<br>• **MAC pause frames**—Number of MAC control frames with **pause** operational code.<br>• **Oversized frames**—There are two possible conditions regarding the number of oversized frames:<br>◦ Packet length exceeds 1518 octets, or<br>◦ Packet length exceeds MRU<br>• **Jabber frames**—Number of frames that were longer than 1518 octets (excluding framing bits, but including FCS octets), and had either an FCS error or an alignment error. This definition of jabber is different from the definition in IEEE-802.3 section 8.2.1.5 (10BASE5) and section 10.3.1.4 (10BASE2). These documents define jabber as the condition in which any packet exceeds 20 ms. The allowed range to detect jabber is from 20 ms to 150 ms.<br>• **Fragment frames**—Total number of packets that were less than 64 octets in length (excluding framing bits, but including FCS octets) and had either an FCS error or an alignment error. Fragment frames normally increment because both runts (which are normal occurrences caused by collisions) and noise hits are counted.<br>• **VLAN tagged frames**—Number of frames that are VLAN tagged. The system uses the TPID of 0x8100 in the frame to determine whether a frame is tagged or not.<br>• **Code violations**—Number of times an event caused the PHY to indicate “Data reception error” or “invalid data symbol error.” | **extensive** |

Table 47: show interfaces Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Filter statistics** | **Receive** and **Transmit** statistics reported by the PIC's MAC address filter subsystem. The filtering is done by the content-addressable memory (CAM) on the PIC. The filter examines a packet's source and destination MAC addresses to determine whether the packet should enter the system or be rejected.<br>• **Input packet count**—Number of packets received from the MAC hardware that the filter processed.<br>• **Input packet rejects**—Number of packets that the filter rejected because of either the source MAC address or the destination MAC address.<br>• **Input DA rejects**—Number of packets that the filter rejected because the destination MAC address of the packet is not on the accept list. It is normal for this value to increment. When it increments very quickly and no traffic is entering the device from the far-end system, either there is a bad ARP entry on the far-end system, or multicast routing is not on and the far-end system is sending many multicast packets to the local device (which the router is rejecting).<br>• **Input SA rejects**—Number of packets that the filter rejected because the source MAC address of the packet is not on the accept list. The value in this field should increment only if source MAC address filtering has been enabled. If filtering is enabled, if the value increments quickly, and if the system is not receiving traffic that it should from the far-end system, it means that the user-configured source MAC addresses for this interface are incorrect.<br>• **Output packet count**—Number of packets that the filter has given to the MAC hardware.<br>• **Output packet pad count**—Number of packets the filter padded to the minimum Ethernet size (60 bytes) before giving the packet to the MAC hardware. Usually, padding is done only on small ARP packets, but some very small IP packets can also require padding. If this value increments rapidly, either the system is trying to find an ARP entry for a far-end system that does not exist or it is misconfigured.<br>• **Output packet error count**—Number of packets with an indicated error that the filter was given to transmit. These packets are usually aged packets or are the result of a bandwidth problem on the FPC hardware. On a normal system, the value of this field should not increment.<br>• **CAM destination filters, CAM source filters**—Number of entries in the CAM dedicated to destination and source MAC address filters. There can only be up to 64 source entries. If source filtering is disabled, which is the default, the values for these fields must be 0. | **extensive** |
| **Autonegotiation information** | Information about link autonegotiation.<br>• **Negotiation status:**<br>  ◦ **Incomplete**—Ethernet interface has the speed or link mode configured.<br>  ◦ **No autonegotiation**—Remote Ethernet interface has the speed or link mode configured, or does not perform autonegotiation.<br>  ◦ **Complete**—Ethernet interface is connected to a device that performs autonegotiation and the autonegotiation process is successful. | **extensive** |
| **Packet Forwarding Engine configuration** | Information about the configuration of the Packet Forwarding Engine:<br>• **Destination slot**—FPC slot number. | **extensive** |

Table 47: show interfaces Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **CoS information** | Information about the CoS queue for the physical interface.<br>• **CoS transmit queue**—Queue number and its associated user-configured forwarding class name.<br>• **Bandwidth %**—Percentage of bandwidth allocated to the queue.<br>• **Bandwidth bps**—Bandwidth allocated to the queue (in bps).<br>• **Buffer %**—Percentage of buffer space allocated to the queue.<br>• **Buffer usec**—Amount of buffer space allocated to the queue, in microseconds. This value is nonzero only if the buffer size is configured in terms of time.<br>• **Priority**—Queue priority: **low** or **high**.<br>• **Limit**—Displayed if rate limiting is configured for the queue. Possible values are **none** and **exact**. If **exact** is configured, the queue transmits only up to the configured bandwidth, even if excess bandwidth is available. If **none** is configured, the queue transmits beyond the configured bandwidth if bandwidth is available. | **extensive** |
| **Interface transmit statistics** | Status of the **interface-transmit-statistics** configuration: Enabled or Disabled. | **detail extensive** |
| **Queue counters (Egress)** | CoS queue number and its associated user-configured forwarding class name.<br>• **Queued packets**—Number of queued packets.<br>• **Transmitted packets**—Number of transmitted packets.<br>• **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism. | **detail extensive** |
| Logical Interface | | |
| **Logical interface** | Name of the logical interface. | All levels |
| **Index** | Index number of the logical interface, which reflects its initialization sequence. | **detail extensive** none |
| **SNMP ifIndex** | SNMP interface index number for the logical interface. | **detail extensive** none |
| **Generation** | Unique number for use by Juniper Networks technical support only. | **detail extensive** |
| **Flags** | Information about the logical interface. | All levels |
| **Encapsulation** | Encapsulation on the logical interface. | All levels |
| **Traffic statistics** | Number and rate of bytes and packets received and transmitted on the specified interface set.<br>• **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface set. The value in this field also includes the Layer 2 overhead bytes for ingress or egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level.<br>• **Input packets, Output packets**—Number of packets received and transmitted on the interface set. | **detail extensive** |

Table 47: show interfaces Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Local statistics** | Number and rate of bytes and packets destined to the device. | **extensive** |
| **Transit statistics** | Number and rate of bytes and packets transiting the switch.<br><br>NOTE: For Gigabit Ethernet intelligent queuing 2 (IQ2) interfaces, the logical interface egress statistics might not accurately reflect the traffic on the wire when output shaping is applied. Traffic management output shaping might drop packets after they are tallied by the **Output bytes** and **Output packets** interface counters. However, correct values display for both of these egress statistics when per-unit scheduling is enabled for the Gigabit Ethernet IQ2 physical interface, or when a single logical interface is actively using a shared scheduler. | **extensive** |
| **Security** | Security zones that interface belongs to. | **extensive** |
| **Flow Input statistics** | Statistics on packets received by flow module. | **extensive** |
| **Flow Output statistics** | Statistics on packets sent by flow module. | **extensive** |
| **Flow error statistics (Packets dropped due to)** | Statistics on errors in the flow module. | **extensive** |
| **Protocol** | Protocol family. | **detail extensive** none |
| **MTU** | Maximum transmission unit size on the logical interface. | **detail extensive** none |
| **Generation** | Unique number for use by Juniper Networks technical support only. | **detail extensive** |
| **Route Table** | Route table in which the logical interface address is located. For example, **0** refers to the routing table inet.0. | **detail extensive** none |
| **Flags** | Information about protocol family flags. . | **detail extensive** |
| **Addresses, Flags** | Information about the address flags.. | **detail extensive** none |
| **Destination** | IP address of the remote side of the connection. | **detail extensive** none |
| **Local** | IP address of the logical interface. | **detail extensive** none |
| **Broadcast** | Broadcast address of the logical interface. | **detail extensive** none |
| **Generation** | Unique number for use by Juniper Networks technical support only. | **detail extensive** |

## Sample Output

### show interfaces Gigabit Ethernet

```
user@host> show interfaces ge-0/0/1
Physical interface: ge-0/0/1, Enabled, Physical link is Down
  Interface index: 135, SNMP ifIndex: 510
  Link-level type: Ethernet, MTU: 1514, Link-mode: Full-duplex, Speed: 1000mbps,

  BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:1f:12:e4:b1:01, Hardware address: 00:1f:12:e4:b1:01
  Last flapped   : 2015-05-12 08:36:59 UTC (1w1d 22:42 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : LINK
  Active defects : LINK
  Interface transmit statistics: Disabled

  Logical interface ge-0/0/1.0 (Index 71) (SNMP ifIndex 514)
    Flags: Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
    Input packets : 0
    Output packets: 0
    Security: Zone: public
    Protocol inet, MTU: 1500
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
        Destination: 1.1.1/24, Local: 1.1.1.1, Broadcast: 1.1.1.255
```

## Sample Output

### show interfaces brief (Gigabit Ethernet)

```
user@host> show interfaces ge-3/0/2 brief
Physical interface: ge-3/0/2, Enabled, Physical link is Up
  Link-level type: 52, MTU: 1522, Speed: 1000mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None

  Logical interface ge-3/0/2.0
    Flags: SNMP-Traps 0x4000
    VLAN-Tag [ 0x8100.512 0x8100.513 ] In(pop-swap 0x8100.530) Out(swap-push
    0x8100.512 0x8100.513)
    Encapsulation: VLAN-CCC
    ccc

  Logical interface ge-3/0/2.32767
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
```

## Sample Output

### show interfaces detail (Gigabit Ethernet)

```
user@host> show interfaces ge-0/0/1 detail
Physical interface: ge-0/0/1, Enabled, Physical link is Down
  Interface index: 135, SNMP ifIndex: 510, Generation: 138
  Link-level type: Ethernet, MTU: 1514, Link-mode: Full-duplex, Speed: 1000mbps,
 BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled, Source filtering:
 Disabled,
  Flow control: Enabled, Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:1f:12:e4:b1:01, Hardware address: 00:1f:12:e4:b1:01
  Last flapped   : 2015-05-12 08:36:59 UTC (1w2d 00:00 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                    0                    0 pps
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 best-effort                    0                    0                    0

    1 expedited-fo                   0                    0                    0

    2 assured-forw                   0                    0                    0

    3 network-cont                   0                    0                    0

  Queue number:         Mapped forwarding classes
    0                   best-effort
    1                   expedited-forwarding
    2                   assured-forwarding
    3                   network-control
  Active alarms  : LINK
  Active defects : LINK
  Interface transmit statistics: Disabled

  Logical interface ge-0/0/1.0 (Index 71) (SNMP ifIndex 514) (Generation 136)
    Flags: Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                    0                    0 bps
     Output bytes  :                    0                    0 bps
     Input  packets:                    0                    0 pps
```

```
    Output packets:                    0                    0 pps
  Security: Zone: public
  Flow Statistics :
  Flow Input statistics :
    Self packets :                     0
    ICMP packets :                     0
    VPN packets :                      0
    Multicast packets :                0
    Bytes permitted by policy :        0
    Connections established :          0
  Flow Output statistics:
    Multicast packets :                0
    Bytes permitted by policy :        0
  Flow error statistics (Packets dropped due to):
    Address spoofing:                  0
    Authentication failed:             0
    Incoming NAT errors:               0
    Invalid zone received packet:      0
    Multiple user authentications:     0
    Multiple incoming NAT:             0
    No parent for a gate:              0
    No one interested in self packets: 0
    No minor session:                  0
    No more sessions:                  0
    No NAT gate:                       0
    No route present:                  0
    No SA for incoming SPI:            0
    No tunnel found:                   0
    No session for a gate:             0
    No zone or NULL zone binding       0
    Policy denied:                     0
    Security association not active:   0
    TCP sequence number out of window: 0
    Syn-attack protection:             0
    User authentication errors:        0
  Protocol inet, MTU: 1500, Generation: 150, Route table: 0
    Flags: Sendbcast-pkt-to-re
    Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
      Destination: 1.1.1/24, Local: 1.1.1.1, Broadcast: 1.1.1.255, Generation:
150
```

## show interfaces statistics st0.0 detail

```
user@host> show interfaces statistics st0.0 detail
  Logical interface st0.0 (Index 71) (SNMP ifIndex 609) (Generation 136)
    Flags: Up Point-To-Point SNMP-Traps Encapsulation: Secure-Tunnel
    Traffic statistics:
     Input  bytes  :         528152756774
     Output bytes  :         575950643520
     Input  packets:          11481581669
     Output packets:          12520666095
    Local statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
     Input  packets:                    0
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                    0            121859888 bps
     Output bytes  :                    0            128104112 bps
     Input  packets:                    0               331141 pps
     Output packets:                    0               348108 pps
```

```
    Security: Zone: untrust
    Allowed host-inbound traffic : any-service bfd bgp dvmrp igmp ldp msdp nhrp
ospf ospf3 pgm pim rip ripng router-discovery rsvp
    sap vrrp
    Flow Statistics :
    Flow Input statistics :
      Self packets :                    0
      ICMP packets :                    0
      VPN packets :                     0
      Multicast packets :               0
      Bytes permitted by policy :       525984295844
      Connections established :         7
    Flow Output statistics:
      Multicast packets :               0
      Bytes permitted by policy :       576003290222
    Flow error statistics (Packets dropped due to):
      Address spoofing:                 0
      Authentication failed:            0
      Incoming NAT errors:              0
      Invalid zone received packet:     0
      Multiple user authentications:    0
      Multiple incoming NAT:            0
      No parent for a gate:             0
      No one interested in self packets: 0
      No minor session:                 0
      No more sessions:                 0
      No NAT gate:                      0
      No route present:                 2000280
      No SA for incoming SPI:           0
      No tunnel found:                  0
      No session for a gate:            0
      No zone or NULL zone binding      0
      Policy denied:                    0
      Security association not active:  0
      TCP sequence number out of window: 0
      Syn-attack protection:            0
      User authentication errors:       0
    Protocol inet, MTU: 9192
    Max nh cache: 0, New hold nh limit: 0, Curr nh cnt: 0, Curr new hold cnt: 0,
 NH drop cnt: 0
    Generation: 155, Route table: 0
      Flags: Sendbcast-pkt-to-re
```

## Sample Output

### show interfaces extensive (Gigabit Ethernet)

```
user@host> show interfaces ge-0/0/1.0 extensive
Physical interface: ge-0/0/1, Enabled, Physical link is Down
  Interface index: 135, SNMP ifIndex: 510, Generation: 138
  Link-level type: Ethernet, MTU: 1514, Link-mode: Full-duplex, Speed: 1000mbps,

  BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Hold-times     : Up 0 ms, Down 0 ms
```

```
Current address: 00:1f:12:e4:b1:01, Hardware address: 00:1f:12:e4:b1:01
Last flapped   : 2015-05-12 08:36:59 UTC (1w1d 22:57 ago)
Statistics last cleared: Never
Traffic statistics:
 Input  bytes  :                    0                    0 bps
 Output bytes  :                    0                    0 bps
 Input  packets:                    0                    0 pps
 Output packets:                    0                    0 pps
Input errors:
  Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0,
  L3 incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0,
  FIFO errors: 0, Resource errors: 0
Output errors:
  Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,

  FIFO errors: 0, HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
Egress queues: 8 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets

  0 best-effort                   0                    0                    0

  1 expedited-fo                  0                    0                    0

  2 assured-forw                  0                    0                    0

  3 network-cont                  0                    0                    0

Queue number:         Mapped forwarding classes
  0                   best-effort
  1                   expedited-forwarding
  2                   assured-forwarding
  3                   network-control
Active alarms  : LINK
Active defects : LINK
MAC statistics:                      Receive         Transmit
  Total octets                             0                0
  Total packets                            0                0
  Unicast packets                          0                0
  Broadcast packets                        0                0
  Multicast packets                        0                0
  CRC/Align errors                         0                0
  FIFO errors                              0                0
  MAC control frames                       0                0
  MAC pause frames                         0                0
  Oversized frames                         0
  Jabber frames                            0
  Fragment frames                          0
  VLAN tagged frames                       0
  Code violations                          0
Filter statistics:
  Input packet count                       0
  Input packet rejects                     0
  Input DA rejects                         0
  Input SA rejects                         0
  Output packet count                                       0
  Output packet pad count                                   0
  Output packet error count                                 0
  CAM destination filters: 2, CAM source filters: 0
Autonegotiation information:
  Negotiation status: Incomplete
Packet Forwarding Engine configuration:
```

```
    Destination slot: 0
  CoS information:
    Direction : Output
    CoS transmit queue               Bandwidth                Buffer Priority
Limit
                              %            bps     %           usec
    0 best-effort            95      950000000    95              0      low
none
    3 network-control         5       50000000     5              0      low
none
  Interface transmit statistics: Disabled

  Logical interface ge-0/0/1.0 (Index 71) (SNMP ifIndex 514) (Generation 136)
    Flags: Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Transit statistics:
     Input  bytes  :                   0                  0 bps
     Output bytes  :                   0                  0 bps
     Input  packets:                   0                  0 pps
     Output packets:                   0                  0 pps
    Security: Zone: public
    Flow Statistics :
    Flow Input statistics :
      Self packets :                    0
      ICMP packets :                    0
      VPN packets :                     0
      Multicast packets :               0
      Bytes permitted by policy :       0
      Connections established :         0
    Flow Output statistics:
      Multicast packets :               0
      Bytes permitted by policy :       0
    Flow error statistics (Packets dropped due to):
      Address spoofing:                 0
      Authentication failed:            0
      Incoming NAT errors:              0
      Invalid zone received packet:     0
      Multiple user authentications:    0
      Multiple incoming NAT:            0
      No parent for a gate:             0
      No one interested in self packets: 0
      No minor session:                 0
      No more sessions:                 0
      No NAT gate:                      0
      No route present:                 0
      No SA for incoming SPI:           0
      No tunnel found:                  0
      No session for a gate:            0
      No zone or NULL zone binding      0
      Policy denied:                    0
      Security association not active:  0
      TCP sequence number out of window: 0
```

```
    Syn-attack protection:             0
    User authentication errors:        0
  Protocol inet, MTU: 1500, Generation: 150, Route table: 0
    Flags: Sendbcast-pkt-to-re
    Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
      Destination: 1.1.1/24, Local: 1.1.1.1, Broadcast: 1.1.1.255,
      Generation: 150
```

## Sample Output

### show interfaces terse

```
user@host> show interfaces terse
    Interface               Admin Link Proto    Local                 Remote
    ge-0/0/0                up    up
    ge-0/0/0.0              up    up   inet     10.209.4.61/18
    gr-0/0/0                up    up
    ip-0/0/0                up    up
    st0                     up    up
    st0.1                   up    ready inet
    ls-0/0/0                up    up
    lt-0/0/0                up    up
    mt-0/0/0                up    up
    pd-0/0/0                up    up
    pe-0/0/0                up    up
    e3-1/0/0                up    up
    t3-2/0/0                up    up
    e1-3/0/0                up    up
    se-4/0/0                up    down
    t1-5/0/0                up    up
    br-6/0/0                up    up
    dc-6/0/0                up    up
    dc-6/0/0.32767          up    up
    bc-6/0/0:1              down  up
    bc-6/0/0:1.0            up    down
    dl0                     up    up
    dl0.0                   up    up   inet
    dsc                     up    up
    gre                     up    up
    ipip                    up    up
    lo0                     up    up
    lo0.16385               up    up   inet     10.0.0.1              --> 0/0
                                                 10.0.0.16             --> 0/0
    lsi                     up    up
    mtun                    up    up
    pimd                    up    up
    pime                    up    up
    pp0                     up    up
```

## Sample Output

### show interfaces controller (Channelized E1 IQ with Logical E1)

```
user@host> show interfaces controller ce1-1/2/6

Controller                                                        Admin Link
ce1-1/2/6                                                         up    up
e1-1/2/6                                                          up    up
```

### show interfaces controller (Channelized E1 IQ with Logical DS0)

```
user@host> show interfaces controller ce1-1/2/3

Controller                                                      Admin Link
ce1-1/2/3                                                       up    up

ds-1/2/3:1                                                      up    up

ds-1/2/3:2                                                      up    up
```

## Sample Output

### show interfaces descriptions

```
user@host> show interfaces descriptions
 Interface       Admin Link Description
 so-1/0/0        up    up   M20-3#1
 so-2/0/0        up    up   GSR-12#1
 ge-3/0/0        up    up   SMB-OSPF_Area300
 so-3/3/0        up    up   GSR-13#1
 so-3/3/1        up    up   GSR-13#2
 ge-4/0/0        up    up   T320-7#1
 ge-5/0/0        up    up   T320-7#2
 so-7/1/0        up    up   M160-6#1
 ge-8/0/0        up    up   T320-7#3
 ge-9/0/0        up    up   T320-7#4
 so-10/0/0       up    up   M160-6#2
 so-13/0/0       up    up   M20-3#2
 so-14/0/0       up    up   GSR-12#2
 ge-15/0/0       up    up   SMB-OSPF_Area100
 ge-15/0/1       up    up   GSR-13#3
```

## Sample Output

### show interfaces destination-class all

```
user@host> show interfaces destination-class all
Logical interface so-4/0/0.0
                                          Packets                 Bytes
     Destination class            (packet-per-second)    (bits-per-second)
                       gold                        0                     0
                                                (   0) (                 0)
                     silver                        0                     0
                                                (   0) (                 0)
  Logical interface so-0/1/3.0
                                          Packets                 Bytes
     Destination class            (packet-per-second)    (bits-per-second)
                       gold                        0                     0
                                                (   0) (                 0)
                     silver                        0                     0
                                                (   0) (                 0)
```

## Sample Output

### show interfaces diagnostics optics

```
user@host> show interfaces diagnostics optics ge-2/0/0
```

```
Physical interface: ge-2/0/0
    Laser bias current                          :  7.408 mA
    Laser output power                          :  0.3500 mW / -4.56 dBm
    Module temperature                          :  23 degrees C / 73 degrees F
    Module voltage                              :  3.3450 V
    Receiver signal average optical power       :  0.0002 mW / -36.99 dBm
    Laser bias current high alarm               :  Off
    Laser bias current low alarm                :  Off
    Laser bias current high warning             :  Off
    Laser bias current low warning              :  Off
    Laser output power high alarm               :  Off
    Laser output power low alarm                :  Off
    Laser output power high warning             :  Off
    Laser output power low warning              :  Off
    Module temperature high alarm               :  Off
    Module temperature low alarm                :  Off
    Module temperature high warning             :  Off
    Module temperature low warning              :  Off
    Module voltage high alarm                   :  Off
    Module voltage low alarm                    :  Off
    Module voltage high warning                 :  Off
    Module voltage low warning                  :  Off
    Laser rx power high alarm                   :  Off
    Laser rx power low alarm                    :  On
    Laser rx power high warning                 :  Off
    Laser rx power low warning                  :  On
    Laser bias current high alarm threshold     :  17.000 mA
    Laser bias current low alarm threshold      :  1.000 mA
    Laser bias current high warning threshold   :  14.000 mA
    Laser bias current low warning threshold    :  2.000 mA
    Laser output power high alarm threshold     :  0.6310 mW / -2.00 dBm
    Laser output power low alarm threshold      :  0.0670 mW / -11.74 dBm
    Laser output power high warning threshold   :  0.6310 mW / -2.00 dBm
    Laser output power low warning threshold    :  0.0790 mW / -11.02 dBm
    Module temperature high alarm threshold     :  95 degrees C / 203 degrees F
    Module temperature low alarm threshold      :  -25 degrees C / -13 degrees F
    Module temperature high warning threshold   :  90 degrees C / 194 degrees F
    Module temperature low warning threshold    :  -20 degrees C / -4 degrees F
    Module voltage high alarm threshold         :  3.900 V
    Module voltage low alarm threshold          :  2.700 V
    Module voltage high warning threshold       :  3.700 V
    Module voltage low warning threshold        :  2.900 V
    Laser rx power high alarm threshold         :  1.2590 mW / 1.00 dBm
    Laser rx power low alarm threshold          :  0.0100 mW / -20.00 dBm
    Laser rx power high warning threshold       :  0.7940 mW / -1.00 dBm
    Laser rx power low warning threshold        :  0.0158 mW / -18.01 dBm
```

## Sample Output

### show interfaces far-end-interval coc12-5/2/0

```
user@host> show interfaces far-end-interval coc12-5/2/0
Physical interface: coc12-5/2/0, SNMP ifIndex: 121
  05:30-current:
    ES-L: 1, SES-L: 1, UAS-L: 0
  05:15-05:30:
    ES-L: 0, SES-L: 0, UAS-L: 0
  05:00-05:15:
    ES-L: 0, SES-L: 0, UAS-L: 0
  04:45-05:00:
```

```
    ES-L: 0, SES-L: 0, UAS-L: 0
  04:30-04:45:
    ES-L: 0, SES-L: 0, UAS-L: 0
  04:15-04:30:
    ES-L: 0, SES-L: 0, UAS-L: 0
  04:00-04:15:
...
```

### show interfaces far-end-interval coc1-5/2/1:1

```
user@host> run show interfaces far-end-interval coc1-5/2/1:1
 Physical interface: coc1-5/2/1:1, SNMP ifIndex: 342
  05:30-current:
    ES-L: 1, SES-L: 1, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
  05:15-05:30:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
  05:00-05:15:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
  04:45-05:00:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
  04:30-04:45:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
  04:15-04:30:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
  04:00-04:15:
```

## Sample Output

### show interfaces filters

```
user@host> show interfaces filters
Interface       Admin Link Proto Input Filter         Output Filter
ge-0/0/0        up    up
ge-0/0/0.0      up    up   inet
                           iso
ge-5/0/0        up    up
ge-5/0/0.0      up    up   any                        f-any
                           inet                       f-inet
                           multiservice
gr-0/3/0        up    up
ip-0/3/0        up    up
mt-0/3/0        up    up
pd-0/3/0        up    up
pe-0/3/0        up    up
vt-0/3/0        up    up
at-1/0/0        up    up
at-1/0/0.0      up    up   inet
                           iso
at-1/1/0        up    down
at-1/1/0.0      up    down inet
                           iso
....
```

## Sample Output

### show interfaces flow-statistics (Gigabit Ethernet)

```
user@host> show interfaces flow-statistics ge-0/0/1.0
```

```
  Logical interface ge-0/0/1.0 (Index 70) (SNMP ifIndex 49)
    Flags: SNMP-Traps Encapsulation: ENET2
    Input packets : 5161
    Output packets: 83
    Security: Zone: zone2
    Allowed host-inbound traffic : bootp bfd bgp  dns dvmrp  ldp msdp nhrp ospf
pgm
    pim rip router-discovery rsvp sap vrrp dhcp finger ftp tftp ident-reset http
 https ike
   netconf ping rlogin rpm rsh snmp snmp-trap ssh telnet traceroute xnm-clear-text
 xnm-ssl
    lsping
    Flow Statistics :
    Flow Input statistics :
      Self packets :                    0
      ICMP packets :                    0
      VPN packets :                     2564
      Bytes permitted by policy :       3478
      Connections established :         1
    Flow Output statistics:
      Multicast packets :               0
      Bytes permitted by policy :       16994
    Flow error statistics (Packets dropped due to):
      Address spoofing:                 0
      Authentication failed:            0
      Incoming NAT errors:              0
      Invalid zone received packet:     0
      Multiple user authentications:    0
      Multiple incoming NAT:            0
      No parent for a gate:             0
      No one interested in self packets: 0
      No minor session:                 0
      No more sessions:                 0
      No NAT gate:                      0
      No route present:                 0
      No SA for incoming SPI:           0
      No tunnel found:                  0
      No session for a gate:            0
      No zone or NULL zone binding      0
      Policy denied:                    0
      Security association not active:  0
      TCP sequence number out of window: 0
      Syn-attack protection:            0
      User authentication errors:       0
    Protocol inet, MTU: 1500
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination:  203.0.113.1/24, Local: 203.0.113.2, Broadcast: 2.2.2.255
```

## Sample Output

### show interfaces interval (Channelized OC12)

```
user@host> show interfaces interval t3-0/3/0:0
Physical interface: t3-0/3/0:0, SNMP ifIndex: 23
  17:43-current:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
    SEFS: 0, UAS: 0
  17:28-17:43:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
```

```
    SEFS: 0, UAS: 0
  17:13-17:28:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
    SEFS: 0, UAS: 0
  16:58-17:13:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
    SEFS: 0, UAS: 0
  16:43-16:58:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
...
Interval Total:
    LCV: 230, PCV: 1145859, CCV: 455470, LES: 0, PES: 230, PSES: 230,
    CES: 230, CSES: 230, SEFS: 230, UAS: 238
```

## show interfaces interval (E3)

```
user@host> show interfaces interval e3-0/3/0
Physical interface: e3-0/3/0, SNMP ifIndex: 23
  17:43-current:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
    SEFS: 0, UAS: 0
  17:28-17:43:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
    SEFS: 0, UAS: 0
  17:13-17:28:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
    SEFS: 0, UAS: 0
  16:58-17:13:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
    SEFS: 0, UAS: 0
  16:43-16:58:
    LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
....
Interval Total:
    LCV: 230, PCV: 1145859, CCV: 455470, LES: 0, PES: 230, PSES: 230,
    CES: 230, CSES: 230, SEFS: 230, UAS: 238
```

## show interfaces interval (SONET/SDH)

```
user@host> show interfaces interval so-0/1/0
Physical interface: so-0/1/0, SNMP ifIndex: 19
  20:02-current:
    ES-S: 0, SES-S: 0, SEFS-S: 0, ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0,
    SES-P: 0, UAS-P: 0
  19:47-20:02:
    ES-S: 267, SES-S: 267, SEFS-S: 267, ES-L: 267, SES-L: 267, UAS-L: 267,
    ES-P: 267, SES-P: 267, UAS-P: 267
  19:32-19:47:
    ES-S: 56, SES-S: 56, SEFS-S: 56, ES-L: 56, SES-L: 56, UAS-L: 46, ES-P: 56,
    SES-P: 56, UAS-P: 46
  19:17-19:32:
    ES-S: 0, SES-S: 0, SEFS-S: 0, ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0,
    SES-P: 0, UAS-P: 0
  19:02-19:17:
.....
```

## Sample Output

### show interfaces load-balancing

```
user@host> show interfaces load-balancing
Interface  State           Last change  Member count
ams0       Up              1d 00:50     2
ams1       Up              00:00:59     2
```

### show interfaces load-balancing detail

```
user@host>show interfaces load-balancing detail
Load-balancing interfaces detail
Interface        : ams0
  State          : Up
  Last change    : 1d 00:51
  Member count   : 2
  Members        :
      Interface    Weight   State
      mams-2/0/0    10        Active
      mams-2/1/0    10        Active
```

## Sample Output

### show interfaces mac-database (All MAC Addresses on a Port)

```
user@host> show interfaces mac-database xe-0/3/3
Physical interface: xe-0/3/3, Enabled, Physical link is Up
  Interface index: 372, SNMP ifIndex: 788
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps, Loopback:
None, Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None

  Logical interface xe-0/3/3.0 (Index 364) (SNMP ifIndex 829)
    Flags: SNMP-Traps 0x4004000 Encapsulation: ENET2
  MAC address        Input frames    Input bytes  Output frames   Output bytes
  00:00:00:00:00:00             1             56              0              0
  00:00:c0:01:01:02       7023810      323095260              0              0
  00:00:c0:01:01:03       7023810      323095260              0              0
  00:00:c0:01:01:04       7023810      323095260              0              0
  00:00:c0:01:01:05       7023810      323095260              0              0
  00:00:c0:01:01:06       7023810      323095260              0              0
  00:00:c0:01:01:07       7023810      323095260              0              0
  00:00:c0:01:01:08       7023809      323095214              0              0
  00:00:c0:01:01:09       7023809      323095214              0              0
  00:00:c0:01:01:0a       7023809      323095214              0              0
  00:00:c0:01:01:0b       7023809      323095214              0              0
  00:00:c8:01:01:02      30424784     1399540064       37448598     1722635508
  00:00:c8:01:01:03      30424784     1399540064       37448598     1722635508
  00:00:c8:01:01:04      30424716     1399536936       37448523     1722632058
  00:00:c8:01:01:05      30424789     1399540294       37448598     1722635508
  00:00:c8:01:01:06      30424788     1399540248       37448597     1722635462
  00:00:c8:01:01:07      30424783     1399540018       37448597     1722635462
  00:00:c8:01:01:08      30424783     1399540018       37448596     1722635416
  00:00:c8:01:01:09       8836796      406492616        8836795      406492570
```

```
  00:00:c8:01:01:0a      30424712     1399536752       37448521     1722631966
  00:00:c8:01:01:0b      30424715     1399536890       37448523     1722632058
  Number of MAC addresses : 21
```

### show interfaces mac-database (All MAC Addresses on a Service)

```
user@host> show interfaces mac-database xe-0/3/3
Logical interface xe-0/3/3.0 (Index 364) (SNMP ifIndex 829)
    Flags: SNMP-Traps 0x4004000 Encapsulation: ENET2
  MAC address        Input frames    Input bytes  Output frames   Output bytes
  00:00:00:00:00:00             1             56              0              0
  00:00:c0:01:01:02       7023810      323095260              0              0
  00:00:c0:01:01:03       7023810      323095260              0              0
  00:00:c0:01:01:04       7023810      323095260              0              0
  00:00:c0:01:01:05       7023810      323095260              0              0
  00:00:c0:01:01:06       7023810      323095260              0              0
  00:00:c0:01:01:07       7023810      323095260              0              0
  00:00:c0:01:01:08       7023809      323095214              0              0
  00:00:c0:01:01:09       7023809      323095214              0              0
  00:00:c0:01:01:0a       7023809      323095214              0              0
  00:00:c0:01:01:0b       7023809      323095214              0              0
  00:00:c8:01:01:02      31016568     1426762128       38040381     1749857526
  00:00:c8:01:01:03      31016568     1426762128       38040382     1749857572
  00:00:c8:01:01:04      31016499     1426758954       38040306     1749854076
  00:00:c8:01:01:05      31016573     1426762358       38040381     1749857526
  00:00:c8:01:01:06      31016573     1426762358       38040381     1749857526
  00:00:c8:01:01:07      31016567     1426762082       38040380     1749857480
  00:00:c8:01:01:08      31016567     1426762082       38040379     1749857434
  00:00:c8:01:01:09       9428580      433714680        9428580      433714680
  00:00:c8:01:01:0a      31016496     1426758816       38040304     1749853984
  00:00:c8:01:01:0b      31016498     1426758908       38040307     1749854122
```

### show interfaces mac-database mac-address

```
user@host> show interfaces mac-database xe-0/3/3 mac-address  00:00:c8:01:01:09
Physical interface: xe-0/3/3, Enabled, Physical link is Up
  Interface index: 372, SNMP ifIndex: 788
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps, Loopback:
None, Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None

  Logical interface xe-0/3/3.0 (Index 364) (SNMP ifIndex 829)
    Flags: SNMP-Traps 0x4004000 Encapsulation: ENET2
  MAC address: 00:00:c8:01:01:09, Type: Configured,
    Input bytes     : 202324652
    Output bytes    : 202324560
    Input frames    : 4398362
    Output frames   : 4398360
  Policer statistics:
  Policer type      Discarded frames    Discarded bytes
  Output aggregate           3992386          183649756
```

## Sample Output

### show interfaces mc-ae

```
user@host> show interfaces mc-ae ae0 unit 512
  Member Links   : ae0
  Local Status   : active
  Peer Status    : active
  Logical Interface        : ae0.512
  Core Facing Interface : Label Ethernet Interface
  ICL-PL         : Label Ethernet Interface
```

### show interfaces media (SONET/SDH)

The following example displays the output fields unique to the **show interfaces media** command for a SONET interface (with no level of output specified):

```
user@host> show interfaces media so-4/1/2
Physical interface: so-4/1/2, Enabled, Physical link is Up
  Interface index: 168, SNMP ifIndex: 495
  Link-level type: PPP, MTU: 4474, Clocking: Internal, SONET mode, Speed: OC48,
  Loopback: None, FCS: 16, Payload scrambler: Enabled
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps 16384
  Link flags     : Keepalives
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 1783 (00:00:00 ago), Output: 1786 (00:00:08 ago)
  LCP state: Opened
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
  mpls: Not-configured
  CHAP state: Not-configured
  CoS queues     : 8 supported
  Last flapped   : 2005-06-15 12:14:59 PDT (04:31:29 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SONET alarms   : None
  SONET defects  : None
  SONET errors:
     BIP-B1: 121, BIP-B2: 916, REI-L: 0, BIP-B3: 137, REI-P: 16747, BIP-BIP2: 0
  Received path trace: routerb so-1/1/2
  Transmitted path trace: routera so-4/1/2
```

## Sample Output

### show interfaces policers

```
user@host> show interfaces policers
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/0/0        up    up
ge-0/0/0.0      up    up   inet
                           iso
gr-0/3/0        up    up
ip-0/3/0        up    up
mt-0/3/0        up    up
pd-0/3/0        up    up
pe-0/3/0        up    up
...
so-2/0/0        up    up
so-2/0/0.0      up    up   inet  so-2/0/0.0-in-policer so-2/0/0.0-out-policer
```

```
                              iso
so-2/1/0        up    down
...
```

### show interfaces policers interface-name

```
user@host> show interfaces policers so-2/1/0
Interface       Admin Link Proto Input Policer         Output Policer
so-2/1/0        up    down
so-2/1/0.0      up    down inet  so-2/1/0.0-in-policer so-2/1/0.0-out-policer
                           iso
                           inet6
```

## Sample Output

### show interfaces queue

The following truncated example shows the CoS queue sizes for queues 0, 1, and 3. Queue 1 has a queue buffer size (guaranteed allocated memory) of 9192 bytes.

```
user@host> show interfaces queue
Physical interface: ge-0/0/0, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 509
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: class0
  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 Buffer Usage:
    Reserved buffer      :             118750000 bytes
  Queue-depth bytes      :
    Current              :                     0
..
..
Queue: 1, Forwarding classes: class1
..
..
 Queue Buffer Usage:
    Reserved buffer      :                  9192 bytes
  Queue-depth bytes      :
    Current              :                     0
..
```

```
..
Queue: 3, Forwarding classes: class3
  Queued:
..
..
Queue Buffer Usage:
    Reserved buffer      :               6250000 bytes
  Queue-depth bytes      :
    Current              :                     0
..
..
```

## Sample Output

### show interfaces redundancy

```
user@host> show interfaces redundancy
Interface  State         Last change  Primary    Secondary  Current status
rsp0       Not present                sp-1/0/0   sp-0/2/0   both down
rsp1       On secondary  1d 23:56     sp-1/2/0   sp-0/3/0   primary down
rsp2       On primary    10:10:27     sp-1/3/0   sp-0/2/0   secondary down
rlsq0      On primary    00:06:24     lsq-0/3/0  lsq-1/0/0  both up
```

### show interfaces redundancy (Aggregated Ethernet)

```
user@host> show interfaces redundancy
Interface  State           Last change  Primary      Secondary    Current status
rlsq0      On secondary     00:56:12     lsq-4/0/0    lsq-3/0/0    both up

ae0
ae1
ae2
ae3
ae4
```

### show interfaces redundancy detail

```
user@host> show interfaces redundancy detail
Interface        : rlsq0
  State          : On primary
  Last change    : 00:45:47
  Primary        : lsq-0/2/0
  Secondary      : lsq-1/2/0
  Current status : both up
  Mode           : hot-standby

Interface        : rlsq0:0
  State          : On primary
  Last change    : 00:45:46
  Primary        : lsq-0/2/0:0
  Secondary      : lsq-1/2/0:0
  Current status : both up
  Mode           : warm-standby
```

## Sample Output

### show interfaces routing brief

```
user@host> show interfaces routing brief
```

```
Interface         State Addresses
so-5/0/3.0        Down  ISO   enabled
so-5/0/2.0        Up    MPLS  enabled
                        ISO   enabled
                        INET  192.168.2.120
                        INET  enabled
so-5/0/1.0        Up    MPLS  enabled
                        ISO   enabled
                        INET  192.168.2.130
                        INET  enabled
at-1/0/0.3        Up    CCC   enabled
at-1/0/0.2        Up    CCC   enabled
at-1/0/0.0        Up    ISO   enabled
                        INET  192.168.90.10
                        INET  enabled
lo0.0             Up    ISO   47.0005.80ff.f800.0000.0108.0001.1921.6800.5061.00
                        ISO   enabled
                        INET  127.0.0.1
fxp1.0            Up
fxp0.0            Up    INET  192.168.6.90
```

### show interfaces routing detail

```
user@host> show interfaces routing detail
so-5/0/3.0
  Index: 15, Refcount: 2, State: Up <Broadcast PointToPoint Multicast> Change:<>

  Metric: 0, Up/down transitions: 0, Full-duplex
  Link layer: HDLC serial line Encapsulation: PPP Bandwidth: 155Mbps
  ISO address (null)
    State: <Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
so-5/0/2.0
  Index: 14, Refcount: 7, State: <Up Broadcast PointToPoint Multicast> Change:<>

  Metric: 0, Up/down transitions: 0, Full-duplex
  Link layer: HDLC serial line Encapsulation: PPP Bandwidth: 155Mbps
  MPLS address (null)
    State: <Up Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4458 bytes
  ISO address (null)
    State: <Up Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
  INET address 192.168.2.120
    State: <Up Broadcast PointToPoint Multicast Localup> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
    Local address: 192.168.2.120
    Destination: 192.168.2.110/32
  INET address (null)
    State: <Up Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
...
```

## Sample Output

### show interfaces routing-instance all

```
user@host> show interfaces terse routing-instance all
Interface   Admin   Link   Proto    Local            Remote Instance
at-0/0/1    up      up     inet     10.0.0.1/24
```

```
ge-0/0/0.0  up      up     inet      192.168.4.28/24          sample-a
at-0/1/0.0  up      up     inet6     fe80::a:0:0:4/64         sample-b
so-0/0/0.0  up      up     inet     10.0.0.1/32
```

## Sample Output

### show interfaces snmp-index

```
user@host> show interfaces snmp-index 33
Physical interface: so-2/1/1, Enabled, Physical link is Down
  Interface index: 149, SNMP ifIndex: 33
  Link-level type: PPP, MTU: 4474, Clocking: Internal, SONET mode, Speed: OC48,
  Loopback: None, FCS: 16, Payload scrambler: Enabled
  Device flags   : Present Running Down
  Interface flags: Hardware-Down Point-To-Point SNMP-Traps 16384
  Link flags     : Keepalives
  CoS queues     : 8 supported
  Last flapped   : 2005-06-15 11:45:57 PDT (05:38:43 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  SONET alarms   : LOL, PLL, LOS
  SONET defects  : LOL, PLL, LOF, LOS, SEF, AIS-L, AIS-P
```

## Sample Output

### show interfaces source-class all

```
user@host> show interfaces source-class all
Logical interface so-0/1/0.0
                                          Packets                 Bytes
     Source class                  (packet-per-second)    (bits-per-second)
                           gold                1928095             161959980
                                (                  889) (              597762)
                         bronze                       0                     0
                                (                    0) (                  0)
                         silver                       0                     0
                                (                    0) (                  0)
  Logical interface so-0/1/3.0
                                          Packets                 Bytes
     Source class                  (packet-per-second)    (bits-per-second)
                           gold                       0                     0
                                (                    0) (                  0)
                         bronze                       0                     0
                                (                    0) (                  0)
                         silver                  116113               9753492
                                (                  939) (              631616)
```

## Sample Output

### show interfaces statistics (Fast Ethernet)

```
user@host> show interfaces fe-1/3/1 statistics
Physical interface: fe-1/3/1, Enabled, Physical link is Up
  Interface index: 144, SNMP ifIndex: 1042
  Description: ford fe-1/3/1
  Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
```

```
  CoS queues     : 4 supported, 4 maximum usable queues
  Current address: 00:90:69:93:04:dc, Hardware address: 00:90:69:93:04:dc
  Last flapped   : 2006-04-18 03:08:59 PDT (00:01:24 ago)
  Statistics last cleared: Never
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Input errors: 0, Output errors: 0
  Active alarms  : None
  Active defects : None
  Logical interface fe-1/3/1.0 (Index 69) (SNMP ifIndex 50)
    Flags: SNMP-Traps Encapsulation: ENET2
    Protocol inet, MTU: 1500
      Flags: Is-Primary, DCU, SCU-in
                                                Packets                 Bytes
      Destination class            (packet-per-second)    (bits-per-second)
                          silver1                      0                     0
                                                  (    0) (                 0)
                          silver2                      0                     0
                                                  (    0) (                 0)
                          silver3                      0                     0
                                                  (    0) (                 0)
      Addresses, Flags: Is-Default Is-Preferred Is-Primary
        Destination: 10.27.245/24, Local: 10.27.245.2,
        Broadcast: 10.27.245.255
    Protocol iso, MTU: 1497
      Flags: Is-Primary
```

## Sample Output

### show interfaces switch-port

```
user@host# show interfaces ge-slot/0/0 switch-port port-number
Port 0, Physical link is Up
        Speed: 100mbps, Auto-negotiation: Enabled
   Statistics:                          Receive         Transmit
          Total bytes                     28437086         21792250
          Total packets                     409145            88008
          Unicast packets                     9987            83817
          Multicast packets                 145002                0
          Broadcast packets                 254156             4191
          Multiple collisions                   23               10
          FIFO/CRC/Align errors                  0                0
          MAC pause frames                       0                0
          Oversized frames                       0
          Runt frames                            0
          Jabber frames                          0
          Fragment frames                        0
          Discarded frames                       0
        Autonegotiation information:
          Negotiation status: Complete
          Link partner:
              Link mode: Full-duplex, Flow control: None, Remote fault: OK, Link
 partner Speed: 100 Mbps
          Local resolution:
              Flow control: None, Remote fault: Link OK
```

## Sample Output

### show interfaces transport pm

```
user@host> show interfaces transport pm all current et-0/1/0
Physical interface: et-0/1/0, SNMP ifIndex 515
  14:45-current          Elapse time:900 Seconds
  Near End       Suspect Flag:False             Reason:None
    PM                  COUNT          THRESHOLD       TCA-ENABLED     TCA-RAISED

  OTU-BBE                 0                800              No              No
  OTU-ES                  0                135              No              No
  OTU-SES                 0                90               No              No
  OTU-UAS                 427              90               No              No
  Far End        Suspect Flag:True              Reason:Unknown
    PM                  COUNT          THRESHOLD       TCA-ENABLED     TCA-RAISED

  OTU-BBE                 0                800              No              No
  OTU-ES                  0                135              No              No
  OTU-SES                 0                90               No              No
  OTU-UAS                 0                90               No              No
  Near End       Suspect Flag:False             Reason:None
    PM                  COUNT          THRESHOLD       TCA-ENABLED     TCA-RAISED

  ODU-BBE                 0                800              No              No
  ODU-ES                  0                135              No              No
  ODU-SES                 0                90               No              No
  ODU-UAS                 427              90               No              No
  Far End        Suspect Flag:True              Reason:Unknown
    PM                  COUNT          THRESHOLD       TCA-ENABLED     TCA-RAISED

  ODU-BBE                 0                800              No              No
  ODU-ES                  0                135              No              No
  ODU-SES                 0                90               No              No
  ODU-UAS                 0                90               No              No
  FEC            Suspect Flag:False             Reason:None
    PM                  COUNT          THRESHOLD       TCA-ENABLED     TCA-RAISED

  FEC-CorrectedErr        2008544300       0                NA              NA
  FEC-UncorrectedWords    0                0                NA              NA
  BER            Suspect Flag:False             Reason:None
    PM                MIN       MAX       AVG       THRESHOLD       TCA-ENABLED
     TCA-RAISED
  BER                 3.6e-5    5.8e-5    3.6e-5    10.0e-3         No
       Yes
Physical interface: et-0/1/0, SNMP ifIndex 515
  14:45-current
  Suspect Flag:True             Reason:Object Disabled
    PM                           CURRENT   MIN       MAX      AVG       THRESHOLD
      TCA-ENABLED       TCA-RAISED
                                                                                 (MIN)
(MAX)    (MIN) (MAX)       (MIN) (MAX)
  Lane chromatic dispersion          0        0         0         0         0
 0        NA    NA         NA     NA
  Lane differential group delay      0        0         0         0         0
 0        NA    NA         NA     NA
    q Value                             120      120       120       120       0
   0        NA    NA         NA     NA
  SNR                                  28       28        29        28        0
 0        NA    NA         NA     NA
```

```
 Tx output power(0.01dBm)          -5000    -5000     -5000     -5000     -300
-100     No    No          No     No
 Rx input power(0.01dBm)           -3642    -3665     -3626     -3637     -1800
-500     No    No          No     No
 Module temperature(Celsius)        46       46        46        46        -5
75       No    No          No     No
 Tx laser bias current(0.1mA)       0        0         0         0         0
0        NA    NA          NA     NA
 Rx laser bias current(0.1mA)       1270     1270      1270      1270      0
0        NA    NA          NA     NA
 Carrier frequency offset(MHz)     -186     -186      -186      -186      -5000
5000     No    No          No     No
```

## Sample Output

### show security zones

```
user@host> show security zones
Functional zone: management
   Description: This is the management zone.
   Policy configurable: No
   Interfaces bound: 1
   Interfaces:
     ge-0/0/0.0
Security zone: Host
   Description: This is the host zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     fxp0.0
Security zone: abc
   Description: This is the abc zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/1.0
Security zone: def
   Description: This is the def zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/2.0
```

## show interfaces swfabx

**Supported Platforms** SRX Series, vSRX

**Syntax** show interfaces (swfab0 | swfab1)

**Release Information** Command introduced in Junos OS Release 11.1.

**Description** Display the configured interfaces for each swfab interface. The swfab interface can contain one or more members because it is an aggregated interface.

**Required Privilege Level** view

**Related Documentation**
- clear interfaces statistics swfabx on page 396

**List of Sample Output**
show interfaces swfab0 on page 458
show interfaces swfab1 on page 458

**Output Fields** Table 48 on page 458 lists the output fields for the **show interfaces <swfab0 | swfab1>** command. Output fields are listed in the approximate order in which they appear.

Table 48: show interfaces <swfab0 | swfab1> Output Fields

| Field Name | Field Description |
|---|---|
| **fabric-options** | The fabric-options hierarchy is configured to be in sync with the fab interfaces. |
| **member-interfaces** | Interfaces specified under member-interfaces are single aggregate interfaces.<br><br>This interface carries internode switching traffic. |

## Sample Output

### show interfaces swfab0

```
user@host# show interfaces swfab0
fabric-options {
       member-interfaces {
           ge-0/0/9;
           ge-0/0/10;
       }
  }
```

### show interfaces swfab1

```
user@host# show interfaces swfab1
fabric-options {
       member-interfaces {
           ge-7/0/9;
```

```
            ge-7/0/10;
        }
    }
```

## show mvrp

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show mvrp

**Release Information** Command introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Display Multiple VLAN Registration Protocol (MVRP) configuration information.

**Required Privilege Level** view

**Related Documentation**

- show mvrp applicant-state on page 462
- show mvrp dynamic-vlan-memberships on page 464
- show mvrp interface on page 466
- show mvrp registration-state on page 468
- show mvrp statistics on page 470

**List of Sample Output** show mvrp on page 461

**Output Fields** Table 49 on page 460 lists the output fields for the **show mvrp** command. Output fields are listed in the approximate order in which they appear.

### Table 49: show mvrp Output Fields

| Field Name | Field Description |
|---|---|
| **MVRP dynamic VLAN creation** | Displays whether global MVRP dynamic VLAN creation is **Enabled** or **Disabled**. |
| **MVRP BPDU MAC address** | Displays the multicast media access control (MAC) address for MVRP. If configured, the provider MVRP multicast MAC address is used; otherwise, the customer MVRP multicast MAC address is used. |
| **MVRP timers (ms)** | Displays MVRP timer information:<br>• **Interface**—The interface on which MVRP is configured.<br>• **Join**—The maximum number of milliseconds the interfaces must wait before sending VLAN advertisements.<br>• **Leave**—The number of milliseconds an interface must wait after receiving a Leave message to remove the interface from the VLAN specified in the message.<br>• **LeaveAll**— The interval at which LeaveAll messages are sent on interfaces. LeaveAll messages maintain current MVRP VLAN membership information in the network. |

## Sample Output

### show mvrp

```
user@host> show mvrp
MVRP configuration for routing instance 'default-switch'
MVRP dynamic VLAN creation : Enabled
MVRP BPDU MAC address      : Customer bridge group (00-00-5E-00-53-00)
MVRP timers (ms)
  Interface       Join   Leave  LeaveAll
  ge-0/0/1        200     800     60
```

## show mvrp applicant-state

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show mvrp applicant-state

**Release Information** Command introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Display Multiple VLAN Registration Protocol (MVRP) applicant state information.

**Required Privilege Level** view

**Related Documentation**

- show mvrp on page 460
- show mvrp interface on page 466
- show mvrp registration-state on page 468
- show mvrp statistics on page 470

**List of Sample Output** show mvrp applicant-state on page 463

**Output Fields** Table 50 on page 462 lists the output fields for the **show mvrp applicant-state** command. Output fields are listed in the approximate order in which they appear.

### Table 50: show mvrp applicant-state Output Fields

| Field Name | Field Description |
|---|---|
| **VLAN Id** | Displays the VLAN ID number. |
| **Interface** | Displays the interface number associated with the VLAN ID. |
| **State** | Displays one of the following MVRP registrar states:<br>• VO— Very anxious observer.<br>• VP —Very anxious passive.<br>• VA —Very anxious new.<br>• AN —Anxious new.<br>• AA —Anxious active.<br>• QA —Quiet active.<br>• LA —Leaving active.<br>• AO —Anxious observer.<br>• QO —Quiet observer.<br>• LO —Leaving observer.<br>• AP —Anxious passive.<br>• QA —Quiet passive. |

## Sample Output

### show mvrp applicant-state

```
user@host> show mvrp applicant-state
MVRP applicant state for routing instance 'default-switch'
 (VO) Very anxious observer, (VP) Very anxious passive, (VA) Very anxious new,
 (AN) Anxious new, (AA) Anxious active, (QA) Quiet active, (LA) Leaving active,
 (AO) Anxious observer, (QO) Quiet observer, (LO) Leaving observer,
 (AP) Anxious passive, (QP) Quiet passive

VLAN Id     Interface          State
    1       ge-0/0/1          Idle (VO)
    30      ge-0/0/1          Idle (VO)
    40      ge-0/0/1          Idle (VO)
    50      ge-0/0/1          Idle (VO)
    100     ge-0/0/1          Idle (VO)
```

## show mvrp dynamic-vlan-memberships

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show mvrp dynamic-vlan-memberships

**Release Information** Command introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Display all VLANs that have been created dynamically using Multiple VLAN Registration Protocol (MVRP) on the SRX Series device.

**Required Privilege Level** clear

**Related Documentation**

- show mvrp on page 460
- show mvrp applicant-state on page 462
- show mvrp interface on page 466
- show mvrp registration-state on page 468
- show mvrp statistics on page 470

**List of Sample Output** show mvrp dynamic-vlan-memberships on page 464

**Output Fields** Table 51 on page 464 lists the output fields for the **show mvrp dynamic-vlan-memberships** command. Output fields are listed in the approximate order in which they appear.

Table 51: show mvrp dynamic-vlan-memberships Output Fields

| Field Name | Field Description |
|---|---|
| **VLAN Id** | The VLAN ID of the dynamically created VLAN. |
| **Interfaces** | The interface or interfaces that are bound to the dynamically created VLAN. |

## Sample Output

### show mvrp dynamic-vlan-memberships

```
user@host> show mvrp dynamic-vlan-memberships
MVRP dynamic vlans for routing instance 'default-switch'
 (s) static vlan, (f) fixed registration

VLAN Id            Interfaces
    1  (s)
    30 (s)
    40 (s)         ge-0/0/1
```

```
 50 (s)        ge-0/0/1
100 (s)        ge-0/0/1 (f)
```

## show mvrp interface

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show mvrp interface

**Release Information** Command introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Display Multiple VLAN Registration Protocol (MVRP) interface-specific information.

**Required Privilege Level** view

**Related Documentation**

- show mvrp on page 460
- show mvrp applicant-state on page 462
- show mvrp dynamic-vlan-memberships on page 464
- show mvrp registration-state on page 468
- show mvrp statistics on page 470

**List of Sample Output** show mvrp interface on page 466

**Output Fields** Table 52 on page 466 lists the output fields for the **show mvrp interface** command. Output fields are listed in the approximate order in which they appear.

Table 52: show mvrp interface Output Fields

| Field Name | Field Description |
|---|---|
| **Interface** | Interface on which MVRP is configured. |
| **Status** | Status of the MVRP: **Enabled** or **Disabled**. |
| **Registration Mode** | Registration for the interface: **Fixed**, **Forbidden**, or **Normal**. |
| **Applicant Mode** | Applicant mode. |

## Sample Output

### show mvrp interface

```
user@host> show mvrp interface
MVRP interface information for routing instance 'default-switch'

Interface       Status      Registration      Applicant
                Mode        Mode              Mode
ge-0/0/1        Enabled     Normal            Normal
```

## show mvrp registration-state

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show mvrp registration-state

**Release Information** Command introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Display Multiple VLAN Registration Protocol (MVRP) registration state information.

**Required Privilege Level** view

**Related Documentation**

- show mvrp on page 460
- show mvrp dynamic-vlan-memberships on page 464
- show mvrp interface on page 466
- show mvrp statistics on page 470

**List of Sample Output** show mvrp registration-state on page 469

**Output Fields** Table 53 on page 468 lists the output fields for the **show mvrp registration-state** command. Output fields are listed in the approximate order in which they appear.

### Table 53: show mvrp registration-state Output Fields

| Field Name | Field Description |
|---|---|
| **VLAN Id** | Displays the VLAN ID number. |
| **Interface** | Displays the interface number associated with the VLAN ID. |
| **Registrar State** | Displays whether the registrar state is Registered or Empty. |
| **Forced State** | Displays whether the forced state is Registered or Empty. |
| **Managed State** | Displays one of the following states:<br>• **fixed**—VLANs always stay in a registered state and are declared as such on all other forwarding ports.<br>• **normal** —VLANs participate in the MVRP protocol and honor incoming join requests normally.<br>• **forbidden** —VLANs ignore the incoming join requests and always stay in an unregistered state. |
| **STP State** | Displays whether the Spanning Tree Protocol (STP) is Blocking or Forwarding. |

## Sample Output

### show mvrp registration-state

```
user@host> show mvrp registration-state
MVRP registration state for routing instance 'default-switch'

VLAN Id   Interface   Registrar   Forced      Managed    STP
                      State       State       State      State
    1     ge-0/0/1    Empty       Empty       Normal     Forwarding
   30     ge-0/0/1    Empty       Empty       Normal     Forwarding
   40     ge-0/0/1    Registered  Registered  Normal     Forwarding
   50     ge-0/0/1    Registered  Registered  Normal     Forwarding
  100     ge-0/0/1    Empty       Registered  Fixed      Forwarding
```

## show mvrp statistics

**Supported Platforms** SRX1500, SRX300, SRX320, SRX340, SRX345, SRX550M

**Syntax** show mvrp statistics

**Release Information** Command introduced in Junos OS Release 15.1X49-D70 for SRX Series devices.

**Description** Display Multiple VLAN Registration Protocol (MVRP) statistics in the form of Multiple Registration Protocol data unit (MRPDU) messages.

**Required Privilege Level** view

**Related Documentation**

- show mvrp on page 460
- show mvrp applicant-state on page 462
- show mvrp dynamic-vlan-memberships on page 464
- show mvrp interface on page 466
- show mvrp registration-state on page 468

**List of Sample Output** show mvrp statistics on page 470

**Output Fields** Table 54 on page 470 lists the output fields for the **show mvrp statistics** command. Output fields are listed in the approximate order in which they appear.

Table 54: show mvrp statistics Output Fields

| Field Name | Field Description |
|---|---|
| **Interface name** | Interface for which MVRP statistics are displayed. |
| **VLAN IDs registered** | Number of VLAN IDs registered. |
| **Sent MVRP PDUs** | Number of MRPDU messages transmitted from the SRX device. |
| **Received MVRP PDUs without error** | Number of MRPDU messages received on the SRX device. |
| **Received MVRP PDUs with error** | Number of invalid MRPDU messages received on the SRX device. |

## Sample Output

### show mvrp statistics

```
user@host> show mvrp statistics
```

```
MVRP statistics for routing instance 'default-switch'

Interface name                 : ge-0/0/1
VLAN IDs registered            : 2
Sent MVRP PDUs                 : 41
Received MVRP PDUs without error: 28
Received MVRP PDUs with error  : 0
Transmitted Join Empty         : 0
Transmitted Leave All          : 20
Received Join In               : 0
Transmitted Join In            : 0
Transmitted Empty              : 114
Transmitted Leave              : 0
Transmitted In                 : 10
Transmitted New                : 0
Received Leave All             : 1
Received Leave                 : 0
Received In                    : 0
Received Empty                 : 67
Received Join Empty            : 24
Received New                   : 0
```

## show oam ethernet connectivity-fault-management adjacencies

**Supported Platforms** SRX Series

**Syntax** show oam ethernet connectivity-fault-management adjacencies <interface-name>

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Display connectivity fault management (CFM) adjacencies such as maintenance association end point (MEP) identifier, interface, state of connectivity check protocol, and expiration time.

**Options** **interface-name**—Dispaly the name of the interface.

**Required Privilege Level** view

**Related Documentation**
- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398

**List of Sample Output** show oam ethernet connectivity-fault- management adjacencies on page 472

**Output Fields** Table 55 on page 472 lists the output fields for the **show oam ethernet connectivity-fault-management adjacencies** command. Output fields are listed in the approximate order in which they appear

Table 55: show oam ethernet connectivity-fault-management adjacencies Output Fields

| Field Name | Field Description |
|---|---|
| Mep-id | MEP identifier. |
| Interface | Interface identifier. |
| State | Indicates whether the connectivity check protocol is up. |
| Timer to Expire | Indicates the expiration time. |

### Sample Output

#### show oam ethernet connectivity-fault- management adjacencies

```
user@host> show oam ethernet connectivity-fault-management adjacencies
Mep-id     Interface        State        Timer to Expire
         101     ge-0/0/4.0         ok                 29
```

## show oam ethernet connectivity-fault-management forwarding-state

**Supported Platforms** SRX Series

**Syntax** show oam ethernet connectivity-fault-management forwarding-state <interface>

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Display the Ethernet Operation, Administration, and Management (OAM) forwarding state for received packets such as interface name, maintenance domain level, maintenance association end point (MEP) direction configured, and next-hop status and index number.

**Options** **<interface>**—Display the Ethernet OAM state for a forwarding instance.

**Required Privilege Level** view

**Related Documentation**

- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398

**List of Sample Output** show oam ethernet connectivity-fault- management forwarding-state on page 475

**Output Fields** Table 56 on page 474 lists the output fields for the **show oam ethernet connectivity-fault-management forwarding-state** command. Output fields are listed in the approximate order in which they appear.

Table 56: show oam ethernet connectivity-fault-management forwarding-state Output Fields

| Field Name | Field Description |
|---|---|
| Interface name | Interface identifier. |
| Level | Maintenance domain level. |
| Direction | MEP direction configured. |
| Filter action | Filter action for messages at the maintenance domain level. |
| Nexthop type | Next-hop type. |
| Nexthop index | Next-hop index number. |

## Sample Output

### show oam ethernet connectivity-fault- management forwarding-state

```
user@host> show oam ethernet connectivity-fault-management forwarding-state interface
Interface name: ge-0/0/1.0 vlan:100
Instance name: INSTANCE_0 bd_vlan_100
  Maintenance domain forwarding state:

  Level    Direction      Filter action     Nexthop      Nexthop
                                            type         index
  0                       Drop              Discard
  1                       Drop              Discard
  2                       Drop              Discard
  3                       Drop              Discard
  4                       Drop              Discard
  5                       Drop              Discard
  6                       Drop              Discard
  7        down           Receive           Receive
```

## show oam ethernet connectivity-fault-management interfaces

**Supported Platforms** SRX Series

**Syntax** show oam ethernet connectivity-fault-management interfaces <interface name>

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Display Ethernet Operation, Administration, and Management (OAM) information for the specified interface such as link status, maintenance domain level configured, maintenance association end point (MEP) identifier, and MEP neighbors count.

**Options** **<interface name>**—Display connectivity fault management (CFM) information for the specified interface.

**Required Privilege Level** view

**Related Documentation**

- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398

**List of Sample Output** show oam ethernet connectivity-fault- management interfaces on page 477

**Output Fields** Table 57 on page 476 lists the output fields for the **show oam ethernet connectivity-fault-management interfaces** command. Output fields are listed in the approximate order in which they appear.

Table 57: show oam ethernet connectivity-fault-management interfaces Output Fields

| Field Name | Field Description |
|---|---|
| Interfaces | Interface identifier. |
| Link | The local link status is up, down, or oam-down. |
| Status | The status is active or inactive. |
| Level | Maintenance domain level configured. |
| MEP Identifier | MEP identifier. |
| Neighbors | Number of MEP neighbors. |

## Sample Output

### show oam ethernet connectivity-fault- management interfaces

```
user@host> show oam ethernet connectivity-fault-management interfaces
Interfaces      Link      Status      Level      MEP      Neighbours
                                                        Identifier

        ge-0/0/1.0      Up        Active      7          1000          0
```

## show oam ethernet connectivity-fault-management mep-database

**Supported Platforms** SRX Series

**Syntax** show oam ethernet connectivity-fault-management mep-database

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Display Ethernet Operation, Administration, and Management (OAM) maintenance association end point (MEP) database information.

**Options** **<local-mep>**—Identifier for the local MEP (1 through 8191).

**maintenance-association** —Name of the maintenance association.

**maintenance-domain** —Name of the maintenance domain.

**remote-mep** —Identifier for the remote MEP (1 through 8191).

**Required Privilege Level** View

**Related Documentation**

- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398

**List of Sample Output** show oam ethernet connectivity-fault- management mep-database on page 480

**Output Fields** Table 58 on page 478 lists the output fields for the **show oam ethernet connectivity-fault-management mep-database** command. Output fields are listed in the approximate order in which they appear.

Table 58: show oam ethernet connectivity-fault-management mep-database Output Fields

| Field Name | Field Description |
|---|---|
| Maintenance domain name | Maintenance domain name. |
| Format (Maintenance domain) | Maintenance domain name format configured. |
| Level | Maintenance domain level configured. |
| Maintenance association name | Maintenance association name. |
| Format (Maintenance association) | Maintenance association name format configured. |
| Continuity-check status | Continuity check status. |

Table 58: show oam ethernet connectivity-fault-management mep-database Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| Interval | Continuity check message (CCM) interval. |
| MEP identifier | MEP identifier. |
| Direction | MEP direction configured. |
| MAC address | MAC address configured for the MEP. |
| Auto-discovery | Indicates whether automatic discovery is enabled or disabled. |
| Priority | Priority used for CCMs and Link Trace Messages (LTMs) transmitted by the MEP. |
| Interface name | Interface identifier. |
| Interface status | Local interface status. |
| Link status | Local link status. |
| Remote MEP not receiving CCM | Indicates that the remote MEP is not receiving CCMs. |
| Erroneous CCM received | Indicates that erroneous CCMs have been received. |
| Cross-connect CCM received | Indicates that cross-connect CCMs have been received. |
| RDI sent by some MEP | Indicates that the remote defect indication (RDI) bit is set in messages that have been received. The absence of the RDI bit in a CCM indicates that the transmitting MEP is receiving CCMs from all configured MEPs. |
| CCMs sent | Number of CCMs transmitted. |
| CCMs received out of sequence | Number of CCMs received out of sequence. |
| LBMs sent | Number of loopback messages (LBMs) sent. |
| Valid in-order LBRs received | Number of loopback response (LBR) messages received that were valid messages and in sequence. |
| Valid out-of-order LBRs received | Number of LBRs received that were valid messages and not in sequence. |
| LBRs received with corrupted data | Number of LBRs received that were corrupted. |
| LBRs sent | Number of LBRs transmitted. |
| LTMs sent | Number of Link Trace Messages (LTMs) transmitted. |
| LTMs received | Number of LTMs received. |

Table 58: show oam ethernet connectivity-fault-management mep-database Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| LTRs sent | Number of Link Trace Replies (LTRs) transmitted. |
| LTRs received | Number of LTRs received. |
| Sequence number of next LTM request | Sequence number of the next LTM request to be transmitted. |
| 1DMs sent | If the MEP is an initiator for a one-way ETH-DM session, then this is the number of one-way delay measurement (1DM) PDU frames sent to the peer MEP in this session.<br><br>For all other cases, this field displays 0. |
| Valid 1DMs received | If the MEP is a receiver for a one-way ETH-DM session, then this is the number of valid 1DM frames received.<br><br>For all other cases, this field displays 0. |
| Invalid 1DMs received | If the MEP is a receiver for a one-way ETH-DM session, then this is the number of invalid 1DM frames received.<br><br>For all other cases, this field displays 0. |
| DMMs sent | If the MEP is an initiator for a two-way ETH-DM session, then this is the number of delay measurement message (DMM) PDU frames sent to the peer MEP in this session. For all other cases, this field displays 0. |
| DMRs sent | If the MEP is a responder for a ETH-DM session, then this is the number of delay measurement reply (DMR) frames sent.<br><br>For all other cases, this field displays 0. |
| Valid DMRs received | If the MEP is an initiator for a two-way ETH-DM session, then this is the number of valid DMRs received.<br><br>For all other cases, this field displays 0. |
| Invalid DMRs received | If the MEP is an initiator for a two-way ETH-DM session, then this is the number of invalid DMRs received.<br><br>For all other cases, this field displays 0. |

## Sample Output

### show oam ethernet connectivity-fault- management mep-database

```
user@host> show oam ethernet connectivity-fault-management mep-database
maintenance-domain Customer1
Maintenance domain name: Customer1, Format: string, Level: 7
  Maintenance association name: Track_vlan_100, Format: string
  Continuity-check status: enabled, Interval: 1s
  MEP identifier: 1000, Direction: down, MAC address: 2001:db8:5E:00:53:00
```

```
  Auto-discovery: disabled, Priority: 0
  Interface name: ge-0/0/1.0, Interface status: Active, Link status: Up
  Defects:
    Remote MEP not receiving CCM                 : no
    Erroneous CCM received                       : no
    Cross-connect CCM received                   : no
    RDI sent by some MEP                         : no
  Statistics:
    CCMs sent                                    : 170114
    CCMs received out of sequence                : 0
    LBMs sent                                    : 0
    Valid in-order LBRs received                 : 0
    Valid out-of-order LBRs received             : 0
    LBRs received with corrupted data            : 0
    LBRs sent                                    : 0
    LTMs sent                                    : 0
    LTMs received                                : 1
    LTRs sent                                    : 1
    LTRs received                                : 0
    Sequence number of next LTM request          : 0
    1DMs sent                                    : 0
    Valid 1DMs received                          : 0
    Invalid 1DMs received                        : 0
    DMMs sent                                    : 0
    DMRs sent                                    : 0
    Valid DMRs received                          : 0
    Invalid DMRs received                        : 0
Remote MEP count: 1
    Identifier    MAC address        State    Interface
       200     2001:db8:c0:01:01:02       ok    ge-0/0/1.0
    Identifier    MAC address        State    Interface  Timer
```

## show oam ethernet connectivity-fault-management mep-statistics

**Supported Platforms** SRX Series

**Syntax**
```
show oam ethernet connectivity-fault-management mep-statistics
count
local-mep
maintenance-association
maintenance-domain
remote-mep
```

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Display Ethernet Operation, Administration, and Management (OAM) maintenance end point (MEP) statistics.

NOTE: The delay measurement statistics are not valid for SRX Series devices, which support only the IEEE 802.1ag standard.

**Options**

**count** —Number of statistics per MEP (1 through 100).

**local-mep** —Identifier for local MEP (1 through 8191).

**maintenance-association**—Name of maintenance association.

**maintenance-domain**—Name of maintenance domain.

**remote-mep** —Identifier for remote MEP (1 through 8191).

**Required Privilege Level** view

**Related Documentation**

- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398

**List of Sample Output** show oam ethernet connectivity-fault- management mep-statistics on page 484

**Output Fields** Table 59 on page 482 lists the output fields for the **show oam ethernet connectivity-fault-management mep-statistics** command. Output fields are listed in the approximate order in which they appear.

Table 59: show oam ethernet connectivity-fault-management mep-statistics Output Fields

| Field Name | Field Description |
|---|---|
| MEP identifier | Maintenance association end point (MEP) identifier. |

Table 59: show oam ethernet connectivity-fault-management mep-statistics Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| CCMs sent | Number of CCMs transmitted. |
| CCMs received out of sequence | Number of CCMs received out of sequence. |
| LBMs sent | Number of loopback messages (LBMs) sent. |
| Valid in-order LBRs received | Number of loopback response (LBR) messages received that were valid messages and in sequence. |
| Valid out-of-order LBRs received | Number of LBRs received that were valid messages and not in sequence. |
| LBRs received with corrupted data | Number of LBRs received that were corrupted. |
| LBRs sent | Number of LBRs transmitted. |
| LTMs sent | Number of Link Trace Messages (LTMs) transmitted. |
| LTMs received | Number of Link Trace Messages received. |
| LTRs sent | Number of Link Trace Replies (LTRs) transmitted. |
| LTRs received | Number of Link Trace responses received. |
| Sequence number of next LTM request | Sequence number of the next Link Trace Message request to be transmitted. |
| 1DMs sent | If the MEP is an initiator in a one-way ETH-DM session, then this is the number of one-way delay measurement (1DM) PDU frames sent to the peer MEP in this session.<br><br>For all other cases, this field displays 0. |
| Valid 1DMs received | If the MEP is a receiver for a one-way ETH-DM session, then this is the number of valid 1DM frames received.<br><br>For all other cases, this field displays 0. |
| Invalid 1DMs received | If the MEP is a receiver for a one-way ETH-DM session, then this is the number of invalid 1DM frames received.<br><br>For all other cases, this field displays 0. |
| DMMs sent | If the MEP is an initiator for a two-way ETH-DM session, then this is the number of delay measurement message (DMM) PDU frames sent to the peer MEP in this session. For all other cases, this field displays 0. |
| DMRs sent | If the MEP is a responder for a ETH-DM session, then this is the number of delay measurement reply (DMR) frames sent. For all other cases, this field displays 0. |

Table 59: show oam ethernet connectivity-fault-management mep-statistics Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| Valid DMRs received | If the MEP is an initiator for a two-way ETH-DM session, then this is the number of valid DMRs received.<br><br>For all other cases, this field displays 0. |
| Invalid DMRs received | If the MEP is an initiator for a two-way ETH-DM session, then this is the number of invalid DMRs received.<br><br>For all other cases, this field displays 0. |

## Sample Output

### show oam ethernet connectivity-fault- management mep-statistics

```
user@host> show oam ethernet connectivity-fault-management mep-statistics
maintenance-domain private maintenance-association private-ma remote-mep 100
MEP identifier: 101, MAC address: 2001:db8:5E:00:53:00
  CCMs sent                                    : 83
  CCMs received out of sequence                : 0
  LBMs sent                                    : 0
  Valid in-order LBRs received                 : 0
  Valid out-of-order LBRs received             : 0
  LBRs received with corrupted data            : 0
  LBRs sent                                    : 0
  LTMs sent                                    : 0
  LTMs received                                : 0
  LTRs sent                                    : 0
  LTRs received                                : 0
  Sequence number of next LTM request          : 0
  1DMs sent                                    : 0
  Valid 1DMs received                          : 0
  Invalid 1DMs received                        : 0
  DMMs sent                                    : 0
  DMRs sent                                    : 0
  Valid DMRs received                          : 0
  Invalid DMRs received                        : 0
```

## show oam ethernet connectivity-fault-management mip

**Supported Platforms** SRX Series

**Syntax**
```
show oam ethernet connectivity-fault-management mip
interface-name
vlan
```

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Display MIP information.

**Options**

**bridge-domain**—Display information for a particular bridge domain.

**instance-name**—Display information for a particular routing instance.

**interface-name**—Display information about the specified logical interface.

**vlan**—Display information about the specified VLAN (1 through 4094).

**Required Privilege Level** view

**Related Documentation**

- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398

**List of Sample Output** show oam ethernet connectivity-fault- management mip on page 485

**Output Fields** Table 60 on page 485 lists the output fields for the **show oam ethernet connectivity-fault-management mip** command. Output fields are listed in the approximate order in which they appear.

Table 60: show oam ethernet connectivity-fault-management mip Output Fields

| Field Name | Field Description |
|---|---|
| Default Maintenance-domain | The default maintenance domain name. |
| Interface | Interface identifier. |
| Level | Maintenance domain level configured. |

### Sample Output

#### show oam ethernet connectivity-fault- management mip

```
user@host> show oam ethernet connectivity-fault-management mip vlan 100
```

```
default maintenance-domain mhf    : default

  Interface       Level
  ge-0/0/1.0          5
  ge-0/0/4.0          5
```

## show oam ethernet connectivity-fault-management path-database

**Supported Platforms** SRX Series

**Syntax**
```
show oam ethernet connectivity-fault-management path-database
<host>
maintenance-association
maintenance-domain
```

**Release Information** Statement introduced in Junos OS Release 12.1X44-D10.

**Description** Display the Link Trace path database for a remote host.

**Options**
**<host>**—MAC address of the remote host in xx:xx:xx:xx:xx:xx format.

**maintenance-association** —Name of the maintenance association.

**maintenance-domain** —Name of the maintenance domain.

**Required Privilege Level** view

**Related Documentation**
- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398

**List of Sample Output** show oam ethernet connectivity-fault- management path-database on page 488

**Output Fields** Table 61 on page 487 lists the output fields for the **show oam ethernet connectivity-fault-management path-database** command. Output fields are listed in the approximate order in which they appear.

Table 61: show oam ethernet connectivity-fault-management path-database Output Fields

| Field Name | Field Description |
|---|---|
| Interface | Interface Identifier. |
| Maintenance Domain | Maintenance domain name. |
| Maintenance Association | Maintenance association name. |
| Level | Maintenance domain level configured for the maintenance domain. |
| Hop | Sequential hop count of the Link Trace path. |
| TTL | Number of hops remaining in the Link Trace message (LTM). The time to live (TTL) is decremented at each hop. |

Table 61: show oam ethernet connectivity-fault-management path-database Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| Source MAC Address | MAC address of the 802.1ag maintenance association intermediate point (MIP) that is forwarding the LTM. |
| Next-hop MAC Address | MAC address of the 802.1ag node that is the next hop in the LTM path. |
| Transaction Identifier | Identifier maintained by the MEP. Each LTM uses a transaction identifier. The transaction identifier is maintained globally across all maintenance domains. Use the transaction identifier to match an incoming Link Trace Reply (LTR) with a previously sent LTM. |

## Sample Output

### show oam ethernet connectivity-fault- management path-database

```
user@host> show oam ethernet connectivity-fault-management path-database
Interface : ge-0/0/4
      Maintenance Domain: private, Level: 5
      Maintenance Association: private-ma, Local Mep: 100

    Hop   TTL    Source MAC address        Next-hop MAC address
    Transaction Identifier:0
    1     63     00:00:5E:00:53:AA        00:00:5E:00:53:AB
    2     62     00:00:5E:00:53:AC         00:00:5E:00:53:AD
    Transaction Identifier:1
    1     63     00:00:5E:00:53:AE         00:00:5E:00:53:AF
    2     62     00:00:5E:00:53:AG         00:00:5E:00:53:AH
    Transaction Identifier:2
    1     63     00:00:5E:00:53:AI         00:00:5E:00:53:AJ
    2     62     00:00:5E:00:53:AK         00:00:5E:00:53:AL
    Transaction Identifier:3
    1     63     00:00:5E:00:53:AM         00:00:5E:00:53:AN
    2     62     00:00:5E:00:53:AO         00:00:5E:00:53:AP
```

## show oam ethernet link-fault-management

**Supported Platforms** SRX Series

**Syntax**
show oam ethernet link-fault-management
<brief | detail>
*<interface-name>*

**Release Information** Statement for SRX Series devices introduced in Junos OS Release 9.5.

**Description** Display Operation, Administration, and Maintenance (OAM) link fault management (LFM) information for Ethernet interfaces.

**Options**
**brief | detail**—(Optional) Display the specified level of output.

***interface-name***—(Optional) Display link fault management information for the specified Ethernet interface only.

**Required Privilege Level** view

**Related Documentation**
- clear oam ethernet connectivity-fault-management path-database on page 397
- clear oam ethernet connectivity-fault-management statistics on page 398
- Understanding Ethernet OAM Link Fault Management for SRX Series Services Gateways on page 299
- Example: Configuring Ethernet OAM Link Fault Management on page 301

**List of Sample Output**
show oam ethernet link-fault-management brief on page 493
show oam ethernet link-fault-management detail on page 493

**Output Fields** Table 62 on page 489 lists the output fields for the **show oam ethernet link-fault-management** command. Output fields are listed in the approximate order in which they appear.

### Table 62: show oam ethernet link-fault-management Output Fields

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Status** | Status of the established link.<br>• **Fail**—A link fault condition exists.<br>• **Running**—A link fault condition does not exist. | All levels |

Table 62: show oam ethernet link-fault-management Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Discovery state** | State of the discovery mechanism:<br>• **Passive Wait**<br>• **Send Any**<br>• **Send Local Remote**<br>• **Send Local Remote Ok** | All levels |
| **Peer address** | Address of the OAM peer. | All levels |
| **Flags** | Information about the interface.<br>• **Remote-Stable**—Indicates remote OAM client acknowledgment of, and satisfaction with, local OAM state information. **False** indicates that remote DTE has either not seen or is unsatisfied with local state information. **True** indicates that remote DTE has seen and is satisfied with local state information.<br>• **Local-Stable**—Indicates local OAM client acknowledgment of, and satisfaction with, remote OAM state information. **False** indicates that local DTE either has not seen or is unsatisfied with remote state information. **True** indicates that local DTE has seen and is satisfied with remote state information.<br>• **Remote-State-Valid**—Indicates the OAM client has received remote state information found within local information TLVs (type, length, values) of received Information OAM PDUs. **False** indicates that the OAM client has not seen remote state information. **True** indicates that the OAM client has seen remote state information. | All levels |
| **Remote loopback status** | An OAM entity can put its remote peer into loopback mode using the Loopback control OAM PDU. In loopback mode, every frame received is transmitted back on the same port (except for OAM PDUs, which are needed to maintain the OAM session). | All levels |
| **Remote entity information** | Remote entity information.<br>• **Remote MUX action**—Indicates the state of the multiplexer functions of the OAM sublayer. Device is forwarding non-OAM PDUs to the lower sublayer or discarding non-OAM PDUs.<br>• **Remote parser action**—Indicates the state of the parser function of the OAM sublayer. Device is forwarding non-OAM PDUs to the higher sublayer, looping back non-OAM PDUs to the lower sublayer, or discarding non-OAM PDUs.<br>• **Discovery mode**—Indicates whether discovery mode is active or inactive.<br>• **Unidirectional mode**—Indicates the ability to operate a link in unidirectional mode for diagnostic purposes.<br>• **Remote loopback mode**—Indicates whether remote loopback is supported or not supported.<br>• **Link events**—Indicates whether interpreting link events is supported or not supported on the remote peer.<br>• **Variable requests**—Indicates whether variable requests are supported or not supported. The Variable Request OAM PDU, is used to request one or more MIB variables from the remote peer. | All levels |
| OAM Receive Statistics | | |

Table 62: show oam ethernet link-fault-management Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Information** | Number of information PDUs received. | **detail** |
| **Event** | Number of loopback control PDUs received. | **detail** |
| **Variable request** | Number of variable request PDUs received. | **detail** |
| **Variable response** | Number of variable response PDUs received. | **detail** |
| **Loopback control** | Number of loopback control PDUs received. | **detail** |
| **Organization specific** | Number of vendor organization specific PDUs received. | **detail** |
| OAM Transmit Statistics | | |
| **Information** | Number of information PDUs transmitted. | **detail** |
| **Event** | Number of event notification PDUs transmitted. | **detail** |
| **Variable request** | Number of variable request PDUs transmitted. | **detail** |
| **Variable response** | Number of variable response PDUs transmitted. | **detail** |
| **Loopback control** | Number of loopback control PDUs transmitted. | **detail** |
| **Organization specific** | Number of vendor organization specific PDUs transmitted. | **detail** |
| OAM Received Symbol Error Event information | | |
| **Events** | Number of symbol error event TLVs that have been received after the OAM sublayer was reset. | **detail** |
| **Window** | Symbol error event window in the received PDU.<br><br>The protocol default value is the number of symbols that can be received in one second on the underlying physical layer. | **detail** |
| **Threshold** | Number of errored symbols in the period required for the event to be generated. | **detail** |
| **Errors in period** | Number of symbol errors in the period reported in the received event PDU. | **detail** |
| **Total errors** | Number of errored symbols that have been reported in received event TLVs after the OAM sublayer was reset.<br><br>Symbol errors are coding symbol errors. | **detail** |
| OAM Received Frame Error Event Information | | |
| **Events** | Number of errored frame event TLVs that have been received after the OAM sublayer was reset. | **detail** |

Table 62: show oam ethernet link-fault-management Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Window** | Duration of the window in terms of the number of 100 ms period intervals. | **detail** |
| **Threshold** | Number of detected errored frames required for the event to be generated. | **detail** |
| **Errors in period** | Number of detected errored frames in the period. | **detail** |
| **Total errors** | Number of errored frames that have been reported in received event TLVs after the OAM sublayer was reset.<br><br>A frame error is any frame error on the underlying physical layer. | **detail** |
| OAM Received Frame Period Error Event Information | | |
| **Events** | Number of frame seconds errors event TLVs that have been received after the OAM sublayer was reset. | **detail** |
| **Window** | Duration of the frame seconds window. | **detail** |
| **Threshold** | Number of frame seconds errors in the period. | **detail** |
| **Errors in period** | Number of frame seconds errors in the period. | **detail** |
| **Total errors** | Number of frame seconds errors that have been reported in received event TLVs after the OAM sublayer was reset. | **detail** |
| OAM Transmitted Symbol Error Event Information | | |
| **Events** | Number of symbol error event TLVs that have been transmitted after the OAM sublayer was reset. | **detail** |
| **Window** | The symbol error event window in the transmitted PDU. | **detail** |
| **Threshold** | Number of errored symbols in the period required for the event to be generated. | **detail** |
| **Errors in period** | Number of symbol errors in the period reported in the transmitted event PDU. | **detail** |
| **Total errors** | Number of errored symbols reported in event TLVs that have been transmitted after the OAM sublayer was reset. | **detail** |
| OAM Transmitted Frame Error Event Information | | |
| **Events** | Number of errored frame event TLVs that have been transmitted after the OAM sublayer was reset. | **detail** |
| **Window** | Duration of the window in terms of the number of 100-ms period intervals. | **detail** |
| **Threshold** | Number of detected errored frames required for the event to be generated. | **detail** |
| **Errors in period** | Number of detected errored frames in the period. | **detail** |

Table 62: show oam ethernet link-fault-management Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Total errors** | Number of errored frames that have been detected after the OAM sublayer was reset. | **detail** |

## Sample Output

### show oam ethernet link-fault-management brief

```
user@host> show oam ethernet link-fault-management brief
Interface: ge-0/0/1
  Status: Running, Discovery state: Send Any
  Peer address: 2001:bd8:00:31
  Flags:Remote-Stable Remote-State-Valid Local-Stable 0x50
  Remote loopback status: Disabled on local port, Enabled on peer port
  Remote entity information:
    Remote MUX action: discarding, Remote parser action: loopback
    Discovery mode: active, Unidirectional mode: unsupported
    Remote loopback mode: supported, Link events: supported
    Variable requests: unsupported
```

### show oam ethernet link-fault-management detail

```
user@host> show oam ethernet link-fault-management detail
  Interface: ge-0/0/1
    Status: Running, Discovery state: Send Any
    Peer address: 2001:bd8:00:31
    Flags:Remote-Stable Remote-State-Valid Local-Stable 0x50
    OAM receive statistics:
      Information: 186365, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 0, Organization specific: 0
    OAM transmit statistics:
      Information: 186347, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 0, Organization specific: 0
    OAM received symbol error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame period error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM transmitted symbol error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM transmitted frame error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    Remote entity information:
      Remote MUX action: forwarding, Remote parser action: forwarding
      Discovery mode: active, Unidirectional mode: unsupported
      Remote loopback mode: supported, Link events: supported
      Variable requests: unsupported
```

## show security flow gate family

**Supported Platforms** SRX Series, vSRX

**Syntax** show security flow gate family (inet | inet6)

**Release Information** Command introduced in Junos OS Release 10.4.

**Description** Display filtered summary of information about existing gates, types of gates, and the maximum allowed number of gates.

**Options**
- inet—Displays IPv4 information.
- inet6—Displays IPv6 gate information.

**Required Privilege Level** view

**Related Documentation**
- *show security flow gate*

**Output Fields** Table 63 on page 494 lists the output fields for the **show security flow gate family** command. Output fields are listed in the approximate order in which they appear.

### Table 63: show security flow gate family Output Fields

| Field Name | Field Description |
|---|---|
| **Valid gates** | Number of valid gates. |
| **Pending gates** | Number of pending gates. |
| **Invalidated gates** | Number of invalid gates. |
| **Gates in other states** | Number of gates in other states. |
| **Total gates** | Total number of gates. |

## Sample Output

```
user@host> show security flow gate family inet6
Hole: 2001:13::8-0-0->2001:12::8-33135-33135

  Translated: ::/0->::/0

  Protocol: tcp

  Application: FTP ALG/79
```

```
  Age: 24 seconds

  Flags: 0x8080

  Zone: zserver

  Reference count: 1

  Resource: 1-2-2


Valid gates: 1

Pending gates: 0

Invalidated gates: 0

Gates in other states: 0

Total gates: 1
```

```
user@host> show security flow gate family inet6 destination-prefix 2001:12::8 or source-prefix
Hole: 2001:13::8-0-0->2001:12::8-33135-33135

  Translated: ::/0->::/0

  Protocol: tcp

  Application: FTP ALG/79

  Age: 26 seconds

  Flags: 0x8080

  Zone: zserver

  Reference count: 1

  Resource: 1-2-2


Valid gates: 1

Pending gates: 0

Invalidated gates: 0

Gates in other states: 0

Total gates: 1
```

## show security flow ip-action

**Supported Platforms** SRX Series, vSRX

**Syntax** **show security flow ip-action** [ <filter> ] [ summary family (inet | inet6) ]

**Release Information** Command introduced in Junos OS Release 10.1. Logical systems option added in Junos OS Release 11.2 . Summary option introduced in Junos OS Release 12.1.

**Description** Display the current IP-action settings, based on filtered options, for IP sessions running on the device.

**Options**

- *filter*—Filter the display based on the specified criteria.

  The following filters display those sessions that match the criteria specified by the filter. Refer to the sample output for filtered output examples.

  **all |** [*filter*]—All active sessions on the device.

  **destination-port** ***destination-port***—Destination port number of the traffic. Range is 1 through 65,535.

  **destination-prefix** ***destination-prefix***—Destination IP prefix or address.

  **family (inet | inet6)** [*filter*]—IPv4 traffic or IPv6-NATPT traffic and filtered options.

  **logical-system** ***logical-system-name*** **| all** [*filter*]—Specified logical system or all logical systems.

  **protocol** ***protocol-name*** **|** ***protocol-number*** [*filter*]—Protocol name or number and filtered options.

  - **ah** or **51**
  - **egp** or **8**
  - **esp** or **50**
  - **gre** or **47**
  - **icmp** or **1**
  - **icmp6** or **58**
  - **ipip** or **4**
  - **ospf** or **89**
  - **pim** or **103**
  - **rsvp** or **46**
  - **sctp** or **132**
  - **tcp** or **6**
  - **udp** or **17**

**root-logical-system** [*filter*]—Default logical system information and filtered options.

**source-port** ***source-port***—Source port number of the traffic. Range is 1 through 65,535.

**source-prefix** ***source-prefix***—Source IP prefix or address of the traffic.

- **summary** —Summary information about IP-action entries.

  **family**—Display summary of IP-action entries by family. This option is used to filter the output.

  - **inet**—Display summary of IPv4 entries.
  - **inet6**—Display summary of IPv6 entries.

**Required Privilege Level** view

**Related Documentation**

- *Juniper Networks Devices Processing Overview*
- clear security flow ip-action on page 399
- *clear security flow session destination-port*

**List of Sample Output**

show security flow ip-action on page 498
show security flow ip-action destination-port on page 499
show security flow ip-action destination-prefix on page 500
show security flow ip-action family inet protocol on page 500
show security flow ip-action family inet logical-system all on page 501
show security flow ip-action source-prefix on page 502
show security flow ip-action summary on page 503
show security flow ip-action summary family inet on page 503
show security flow ip-action summary family inet6 on page 503

**Output Fields** Table 64 on page 497 lists the output fields for the **show security flow ip-action** command. Output fields are listed in the approximate order in which they appear.

Table 64: show security flow ip-action Output Fields

| Field Name | Field Description |
|---|---|
| **Src-Addr** | Source address of outbound IP traffic. |
| **Src-Port** | Source port number of outbound IP traffic. |
| **Dst-Addr** | Destination address of inbound IP traffic. |
| **Dst-Port/Proto** | Destination port number and protocol type of inbound IP traffic. |
| **Timeout (sec)** | Configured timeouts and time remaining for an IP session. |
| **Zone** | Security zone associated with an IP session. |

Table 64: show security flow ip-action Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| **Action** | Configured action type, for example, block, close, and notify. |
| **State** | The active mode and passive mode describe the states of the **ip-action** entry. |
| **IPv4 action count** | The total number of IPv4 entries. |
| **IPv6 action count** | The total number of IPv6 entries. |

## Sample Output

### show security flow ip-action

```
user@host> show security flow ip-action
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
 203.0.113.1       *       203.0.113.4         21/tcp         293/300      *
      close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *      203.0.113.4          21/tcp         293/300      *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *      203.0.113.4          21/tcp         293/300      *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *       203.0.113.4          21/tcp         293/300      *
  close        Passive
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *      203.0.113.4          21/tcp         293/300       *
    close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *    203.0.113.4          21/tcp         292/300      *
   close        Passive
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
 203.0.113.1         *    203.0.113.4          21/tcp         292/300      *
   close        Active
 IPv4 action count: 1 on FPC1.PIC3
```

```
 IPv4 action count: Active mode 1 on all PICs
 IPv6 action count: 0 on FPC0.PIC1
 IPv6 action count: 0 on FPC0.PIC2
 IPv6 action count: 0 on FPC0.PIC3
 IPv6 action count: 0 on FPC1.PIC0
 IPv6 action count: 0 on FPC1.PIC1
 IPv6 action count: 0 on FPC1.PIC2
 IPv6 action count: 0 on FPC1.PIC3
 IPv6 action count: Active mode 0 on all PICs
```

## show security flow ip-action destination-port

```
user@host> show security flow ip-action destination-port 21

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4        21/tcp         274/300      *
  close        Passive
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
  close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1        *        203.0.113.4         21/tcp         274/300      *
  close        Passive
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         273/300      *
  close        Active
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
 IPv6 action count: 0 on FPC0.PIC1
 IPv6 action count: 0 on FPC0.PIC2
 IPv6 action count: 0 on FPC0.PIC3
 IPv6 action count: 0 on FPC1.PIC0
 IPv6 action count: 0 on FPC1.PIC1
```

```
 IPv6 action count: 0 on FPC1.PIC2
 IPv6 action count: 0 on FPC1.PIC3
 IPv6 action count: Active mode 0 on all PICs
```

### show security flow ip-action destination-prefix

```
user@host> show security flow ip-action destination-prefix 203.0.113.4/8

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
192.0.2.3           *        203.0.113.4         21/tcp         245/300      *
 close        Passive
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
192.0.2.3           *        203.0.113.4         21/tcp         245/300      *
 close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
  close        Passive
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
 close        Active
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

### show security flow ip-action family inet protocol

```
user@host> show security flow ip-action family inet protocoludp
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
        close        Passive
 IPv4 action count: 1 on FPC0.PIC1
```

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300     *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
    close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
     close        Active
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
     close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1        *        203.0.113.4         69/udp         287/300      *
    close        Passive
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
    close        Passive
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

## show security flow ip-action family inet logical-system all

```
user@host> show security flow ip-action family inet logical-system all

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
        close        Passive           root-logical-system
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
 close        Passive           root-logical-system
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *       203.0.113.4         69/udp         267/300      *
  close        Passive           root-logical-system
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
```

```
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
  close        Active            root-logical-system
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
  close        Passive           root-logical-system
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4        69/udp         266/300      *
  close        Passive           root-logical-system
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4        69/udp         266/300      *
  close        Passive           root-logical-system
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

## show security flow ip-action source-prefix

```
user@host> show security flow ip-action source-prefix 192.0.2.3/8

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
   close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4        69/udp         244/300      *
    close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
     close        Active
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
    close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
     close        Passiveo
```

```
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
     close        Passive
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

## show security flow ip-action summary

```
user@host> show security flow ip-action summary

 IPv4 action count: 1 on FPC0.PIC1
 IPv4 action count: 1 on FPC0.PIC2
 IPv4 action count: 1 on FPC0.PIC3
 IPv4 action count: 1 on FPC1.PIC0
 IPv4 action count: 1 on FPC1.PIC1
 IPv4 action count: 1 on FPC1.PIC2
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
 IPv6 action count: 0 on FPC0.PIC1
 IPv6 action count: 0 on FPC0.PIC2
 IPv6 action count: 0 on FPC0.PIC3
 IPv6 action count: 0 on FPC1.PIC0
 IPv6 action count: 0 on FPC1.PIC1
 IPv6 action count: 0 on FPC1.PIC2
 IPv6 action count: 0 on FPC1.PIC3
 IPv6 action count: Active mode 0 on all PICs
```

## show security flow ip-action summary family inet

```
user@host> show security flow ip-action summary inet

 IPv4 action count: 1 on FPC0.PIC1
 IPv4 action count: 1 on FPC0.PIC2
 IPv4 action count: 1 on FPC0.PIC3
 IPv4 action count: 1 on FPC1.PIC0
 IPv4 action count: 1 on FPC1.PIC1
 IPv4 action count: 1 on FPC1.PIC2
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

## show security flow ip-action summary family inet6

```
user@host> show security flow ip-action summary family inet6

 IPv6 action count: 1 on FPC0.PIC1
 IPv6 action count: 1 on FPC0.PIC2
 IPv6 action count: 1 on FPC0.PIC3
 IPv6 action count: 1 on FPC1.PIC0
 IPv6 action count: 1 on FPC1.PIC1
 IPv6 action count: 1 on FPC1.PIC2
 IPv6 action count: 1 on FPC1.PIC3
 IPv6 action count: Active mode 1 on all PICs
```

## show security flow session family

**Supported Platforms** SRX Series, vSRX

**Syntax**
show security flow session family (inet | inet6)
[brief | extensive | summary]

**Release Information** Command introduced in Junos OS Release 10.2.

**Description** Display filtered summary of information about existing sessions, including types of sessions, active and failed sessions, and the maximum allowed number of sessions.

**Options**
- **inet**—Display details summary of IPv4 sessions.
- **inet6**—Display details summary of IPv6 sessions.
- brief | extensive | summary–Display the specified level of output.

**Required Privilege Level** view

**Related Documentation**
- *Juniper Networks Devices Processing Overview*
- clear security flow session family on page 401

**List of Sample Output**
show security flow session family inet on page 505
show security flow session family inet brief on page 506
show security flow session family inet extensive on page 506
show security flow session family inet summary on page 508

**Output Fields** Table 65 on page 504 lists the output fields for the **show security flow session family** command. Output fields are listed in the approximate order in which they appear.

### Table 65: show security flow session family Output Fields

| Field Name | Field Description |
|---|---|
| **Session ID** | Number that identifies the session. Use this ID to get more information about the session. |
| **Policy name** | Policy that permitted the traffic. |
| **Timeout** | Idle timeout after which the session expires. |
| **In** | Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes). |

Table 65: show security flow session family Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| **Out** | Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes). |
| **Total sessions** | Total number of sessions. |
| **Status** | Session status. |
| **Flag** | Internal flag depicting the state of the session, used for debugging purposes. |
| **Policy name** | Name and ID of the policy that the first packet of the session matched. |
| **Source NAT pool** | The name of the source pool where NAT is used. |
| **Application** | Name of the application. |
| **Maximum timeout** | Maximum session timeout. |
| **Current timeout** | Remaining time for the session unless traffic exists in the session. |
| **Session State** | Session state. |
| **Start time** | Time when the session was created, offset from the system start time. |
| **Unicast-sessions** | Number of unicast sessions. |
| **Multicast-sessions** | Number of multicast sessions. |
| **Failed-sessions** | Number of failed sessions. |
| **Sessions-in-use** | Number of sessions in use.<br>• Valid sessions<br>• Pending sessions<br>• Invalidated sessions<br>• Sessions in other states |
| **Maximum-sessions** | Number of maximum sessions. |

## Sample Output

### show security flow session family inet

```
root> show security flow session family inet
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
```

```
Session ID: 420000107, Policy name: default-policy-00/2, Timeout: 4, Valid
  In: 203.0.113.0/3 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000202
  Out: 203.0.113.4/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000202
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000115, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000110
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000110

Session ID: 430000117, Policy name: default-policy-00/2, Timeout: 4, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000111
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000111
Total sessions: 2
```

## show security flow session family inet brief

```
root> show security flow session family inet brief
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000115, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/3 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000206
  Out: 203.0.113.4/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000206

Session ID: 420000117, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000207
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000207
Total sessions: 2

Flow Sessions on FPC10 PIC3:

Session ID: 430000119, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000112
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000112
Total sessions: 1
```

## show security flow session family inet extensive

```
root> show security flow session family inet extensive
Flow Sessions on FPC10 PIC1:

Session ID: 410000111, Status: Normal
Flags: 0x80400040/0x0/0x2800023
```

```
Policy name: default-policy-00/2
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 4, Current timeout: 4
Session State: Valid
Start time: 76455, Duration: 0
   In: 203.0.113.0/24 --> 203.0.113.1/24;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0xa0010, Gateway: 203.0.113.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 410000242
   Out: 203.0.113.1/24 --> 203.0.113.10/4;icmp,
    Interface: .local..0,
    Session token: 0x2, Flag: 0x40000030
    Route: 0xfffb0006, Gateway: 203.0.113.1, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 410000242
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Session ID: 420000123, Status: Normal
Flags: 0x80400040/0x0/0x2800023
Policy name: default-policy-00/2
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 4, Current timeout: 2
Session State: Valid
Start time: 76454, Duration: 2
   In: 203.0.113.10/24 --> 203.0.113.11/24;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0xa0010, Gateway: 20010, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 420000210
   Out: 203.0.113.11/24 --> 203.0.113.12/24;icmp,
    Interface: .local..0,
    Session token: 0x2, Flag: 0x40000030
    Route: 0xfffb0006, Gateway: 203.0.113.1, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 420000210
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000131, Status: Normal
Flags: 0x80400040/0x0/0x2800023
Policy name: default-policy-00/2
```

```
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 4, Current timeout: 4
Session State: Valid
Start time: 76421, Duration: 1
   In: 203.0.113.10/24 --> 203.0.113.11/24;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0xa0010, Gateway: 203.0.113.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 430000118
   Out: 203.0.113.12/24 --> 203.0.113.13/24;icmp,
    Interface: .local..0,
    Session token: 0x2, Flag: 0x40000030
    Route: 0xfffb0006, Gateway: 203.0.113.1, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 430000118
Total sessions: 1
```

### show security flow session family inet summary

```
root> show security flow session family inet summary
Flow Sessions on FPC10 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 2
Sessions in other states: 0
Total sessions: 4

Flow Sessions on FPC10 PIC2:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 2
Sessions in other states: 0
Total sessions: 4

Flow Sessions on FPC10 PIC3:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 2
Sessions in other states: 0
Total sessions: 4
```

## show security flow statistics

**Supported Platforms** SRX Series, vSRX

**Syntax**

**show security flow statistics**
<node (*node-id* | all | local | primary)>

**Release Information** Command introduced in Junos OS Release 10.2. Fragmentation counters options introduced in Junos OS Release 15.1X49-90.

**Description** Display security flow statistics on a specific SPU. A flow is a stream of related packets that meet the same matching criteria and share the same characteristics.

A packet undergoes flow-based processing after packet-based filters and some screens have been applied to it. A System Processing Unit (SPU) processes the packets of a flow according to the security features and other services configured for the session. Flow-based packet processing treats related packets, or a stream of packets, in the same way. Packet treatment depends on characteristics that were established for the first packet of the packet stream.

The **show security flow statistics** command displays information for individual SPUs. For each SPU, it shows the number of active sessions on the SPU, the number of packets processed and forwarded, number of packets dropped, the number of packet fragments received in a flow on the SPU, the number of pre-fragmented packets generated, and the number of post-fragmented packets generated.

There are many conditions that can cause a packet to be dropped. Here are some of them:

- A screen module detects IP spoofing
- The IPSec Encapsulating Security Payload (ESP) or the Authentication Header (AH) authentication failed. For example, incoming NAT errors could cause this to happen.
- A packet matches more than one security policy that specifies user authentication. (Sometimes packets are looped through the system more than once. Each time a packet passes through the system, that packet must be permitted by a policy.)
- A time constraint setting expires. For example, multicast streams with a packet interval of more than 60 seconds would experience premature aging-out of flow sessions. (In most cases, you can configure higher time-out value to prevent packet drop.)

Packet fragmentation can occur for a number of reasons, and, in some cases, it can be controlled through a configuration setting. Every link has a maximum transmission unit (MTU) size that specifies the size of the largest packet that the link can transmit. A larger MTU size means that fewer packets are required to transmit a certain amount of data. However, for a packet to successfully traverse the path from the source node to the destination node, the MTU size of the source node egress interface must be no larger than that of the smallest MTU size of all nodes on the path between the source and destination. This value is referred to as the path maximum transmission unit (path MTU).

When a packet is larger that the MTU size on any link in the data path, the link might fragment it or drop it.

- For IPv4, if a node within the path between a source node and a destination node receives a packet that is larger than its MTU size, it can fragment the packet and transmit the resulting smaller packets.
- For IPv6, an intermediate node cannot fragment a packet. If a packet is larger than a link's MTU size, it is likely that the link will drop it. However, the source node (the node that sent the packet) can fragment a packet, and this is done to accommodate a path MTU size-adjustment requirement. Nodes along the path of a packet cannot fragment the packet to transmit it.

The fragmentation counters feature for IPsec tunnels provides the show output information for the pre-fragments generated and post-fragments generated fields.

Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) are introduced for SRX5400, SRX5600, and SRX5800 devices that perform hash-based datapath packet forwarding to interconnect with all existing IOC and SPC cards using the XL chip (packet-processing chip). The IOC3 XL chip uses a hash-based method to distribute ingress traffic to a pool of SPUs by default.

**Options**

- none—Display the security flow statistics information.
- **node**—(Optional) For chassis cluster configurations, display all security flow statistics on a specific node (device) in the cluster.
  - ***node-id***—Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

**Required Privilege Level**

view

**Related Documentation**

- *Juniper Networks Devices Processing Overview*

**List of Sample Output**

show security flow statistics on page 511

**Output Fields**

Table 66 on page 510 lists the output fields for the **show security flow statistics** command. Output fields are listed in the approximate order in which they appear.

### Table 66: show security flow statistics Output Fields

| Field Name | Field Description |
|---|---|
| Current sessions | Number of active sessions on the SPU. |

Table 66: show security flow statistics Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| Packets forwarded | Number of packets received in a security flow of a specific SPU. The packets are processed and forwarded on that SPU. |
| Packets dropped | Number of packets dropped in a flow on a specific SPU.<br><br>The packets are received in the flow. However, during processing, the system discovered sanity check errors, security violations, or other conditions that caused the packet to be dropped.<br><br>See the description for some of the conditions and events that can cause a packet to be dropped. |
| Fragment packets | Number of fragments received in a flow on the SPU. See the description for information about packet fragments. |
| Pre fragments generated | For IPsec tunnels, the number of fragments that are self-generated by the SRX Series device before it encapsulates the packet with the IPsec encryption header. |
| Post fragments generated | For IPsec tunnels, the number of fragments that are received by the SRX Series device and packets that are fragmented after encryption. |

## Sample Output

### show security flow statistics

```
user@host> show security flow statistics
node0:
--------------------------------------------------------------------------

  Flow Statistics of FPC0 PIC1:
    Current sessions: 0
    Packets forwarded: 71186468
    Packets dropped: 2803
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

  Flow Statistics of FPC0 PIC2:
    Current sessions: 0
    Packets forwarded: 71162279
    Packets dropped: 2390
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

  Flow Statistics of FPC0 PIC3:
```

```
  Current sessions: 0
  Packets forwarded: 71147384
  Packets dropped: 2076
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC4 PIC0:
  Current sessions: 0
  Packets forwarded: 71190678
  Packets dropped: 2435
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC4 PIC1:
  Current sessions: 0
  Packets forwarded: 71106410
  Packets dropped: 2209
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC4 PIC2:
  Current sessions: 0
  Packets forwarded: 71254091
  Packets dropped: 2907
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC4 PIC3:
  Current sessions: 0
  Packets forwarded: 71253962
  Packets dropped: 1908
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC5 PIC0:
  Current sessions: 0
  Packets forwarded: 71092466
  Packets dropped: 2700
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC5 PIC1:
  Current sessions: 0
  Packets forwarded: 71385485
  Packets dropped: 3374
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC5 PIC2:
  Current sessions: 0
  Packets forwarded: 71044650
  Packets dropped: 2085
  Fragment packets: 0
  Pre fragments generated: 0
```

```
    Post fragments generated: 0

  Flow Statistics of FPC5 PIC3:
    Current sessions: 0
    Packets forwarded: 71262479
    Packets dropped: 3100
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

  Flow Statistics Summary:
    System total valid sessions: 0
    Packets forwarded: 783086352
    Packets dropped: 27987
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

node1:
--------------------------------------------------------------------------

  Flow Statistics of FPC0 PIC1:
    Current sessions: 0
    Packets forwarded: 0
    Packets dropped: 0
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

  Flow Statistics of FPC0 PIC2:
    Current sessions: 0
    Packets forwarded: 0
    Packets dropped: 0
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

  Flow Statistics of FPC0 PIC3:
    Current sessions: 0
    Packets forwarded: 0
    Packets dropped: 0
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

  Flow Statistics of FPC4 PIC0:
    Current sessions: 0
    Packets forwarded: 0
    Packets dropped: 0
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

  Flow Statistics of FPC4 PIC1:
    Current sessions: 0
    Packets forwarded: 0
    Packets dropped: 0
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0
```

```
Flow Statistics of FPC4 PIC2:
  Current sessions: 0
  Packets forwarded: 0
  Packets dropped: 0
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC4 PIC3:
  Current sessions: 0
  Packets forwarded: 0
  Packets dropped: 0
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC5 PIC0:
  Current sessions: 0
  Packets forwarded: 0
  Packets dropped: 0
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC5 PIC1:
  Current sessions: 0
  Packets forwarded: 0
  Packets dropped: 0
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC5 PIC2:
  Current sessions: 0
  Packets forwarded: 0
  Packets dropped: 0
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics of FPC5 PIC3:
  Current sessions: 0
  Packets forwarded: 0
  Packets dropped: 0
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0

Flow Statistics Summary:
  System total valid sessions: 0
  Packets forwarded: 0
  Packets dropped: 0
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0
```

## show security flow status

**Supported Platforms** SRX Series, vSRX

**Syntax** **show security flow status**

**Release Information** Command introduced in Junos OS Release 10.2; session distribution mode option added in Junos OS Release 12.1X44-D10; enhanced route scaling mode option added in Junos OS Release 12.1X45-D10. GTP-U distribution option added in Junos OS Release 15.1X49-D40.
Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) are introduced for SRX5400, SRX5600, and SRX5800 devices that perform hash-based data path packet forwarding to interconnect with all existing IOC and SPC cards using the XL chip (packet-processing chip).
The IOC3 XL chip uses a hash-based method to distribute ingress traffic to a pool of SPUs by default. Selection of hash keys depends on application protocols.

**Description** Display the flow processing modes and logging status.

**Required Privilege Level** view

**Related Documentation**
- *Juniper Networks Devices Processing Overview*

**List of Sample Output**
show security flow status on page 516
show security flow status (IPsec Performance Acceleration) on page 516
show security flow status (for hash-based datapath forwarding using SRX5K-MPC3-40G10G (IOC3) and SRX5K-MPC3-100G10G (IOC3) on page 517

**Output Fields** Table 67 on page 515 lists the output fields for the **show security flow status** command. Output fields are listed in the approximate order in which they appear.

### Table 67: show security flow status Output Fields

| Field Name | Field Description |
|---|---|
| Flow forwarding mode | Flow processing mode.<br>• Inet forwarding mode<br>• Inet6 forwarding mode<br>• MPLS forwarding mode<br>• ISO forwarding mode<br>• Session distribution mode<br>• Enhanced route scaling mode |

Table 67: show security flow status Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| Flow trace status | Flow logging status.<br>• Flow tracing status<br>• Flow tracing options |
| flow session distribution | SPU load distribution mode.<br>• RR-based<br>• Hash-based<br>GTP-U distribution<br>• Enabled |
| Flow packet ordering | packet-ordering mode.<br>• Hardware<br>• Software |
| Flow ipsec performance acceleration | IPsec VPN performance acceleration status. |

## Sample Output

### show security flow status

```
root> show security flow status
Flow forwarding mode:
Inet forwarding mode: flow based
Inet6 forwarding mode: flow based
MPLS forwarding mode: drop
ISO forwarding mode: drop
Enhanced route scaling mode: Enabled (reboot needed to disable)
Flow trace status
Flow tracing status: on
Flow tracing options: all
Flow session distribution
Distribution mode: Hash-based
GTP-U distribution: Enabled
Flow packet ordering
Ordering mode: Software (reboot needed to change to software)
```

### show security flow status (IPsec Performance Acceleration)

```
root> show security flow status
Flow forwarding mode:
    Inet forwarding mode: flow based
    Inet6 forwarding mode: drop
    MPLS forwarding mode: drop
    ISO forwarding mode: drop
Flow trace status
    Flow tracing status: off
Flow session distribution
    Distribution mode: RR-based
    GTP-U distribution: Enabled Flow packet ordering
```

```
Ordering mode: Software (reboot needed to change to software)
Flow ipsec performance acceleration: on
```

### show security flow status (for hash-based datapath forwarding using SRX5K-MPC3-40G10G (IOC3) and SRX5K-MPC3-100G10G (IOC3)

```
root> show security flow status
node0:
--------------------------------------------------------------------------
  Flow forwarding mode:
    Inet forwarding mode: flow based
    Inet6 forwarding mode: drop
    MPLS forwarding mode: drop
    ISO forwarding mode: drop
  Flow trace status
    Flow tracing status: off
  Flow session distribution
    Distribution mode: Hash-based
    GTP-U distribution: Enabled
  Flow ipsec performance acceleration: off
  Flow packet ordering
    Ordering mode: Hardware

node1:
--------------------------------------------------------------------------
  Flow forwarding mode:
    Inet forwarding mode: flow based
    Inet6 forwarding mode: drop
    MPLS forwarding mode: drop
    ISO forwarding mode: drop
  Flow trace status
    Flow tracing status: off
  Flow session distribution
    Distribution mode: Hash-based
    GTP-U distribution: Enabled
  Flow ipsec performance acceleration: off
  Flow packet ordering
    Ordering mode: Hardware
```

## show security forward-options secure-wire

**Supported Platforms** SRX Series, vSRX

**Syntax** show security forward-options secure-wire <*secure-wire-name*>

**Release Information** Command introduced in Junos OS Release 12.3X48-D10.

**Description** Display information about secure wire mappings.

**Options**

- none—Display information about all configured secure wire mappings.
- *secure-wire-name*—(Optional) Display information about the specified secure wire mapping.

**Required Privilege Level** view

**Related Documentation**

- Understanding Secure Wire on Security Devices on page 75

**List of Sample Output**

show security forward-options secure-wire on page 518
show security forward-options secure-wire pw1 on page 519

**Output Fields** Table 68 on page 518 lists the output fields for the **show security forward-options secure-wire** command. Output fields are listed in the approximate order in which they appear.

Table 68: show security forward-options secure-wire Output Fields

| Field Name | Field Description |
|---|---|
| Secure wire | Name of the secure wire mapping. |
| Interface | One of the peer interfaces in the secure wire mapping. |
| Link | Operational status of the interface link. |
| Interface | The second peer interface in the secure wire mapping. |
| Link | Operational status of the interface link. |

### Sample Output

#### show security forward-options secure-wire

```
user@host> show security forward-options secure-wire
Secure wire                    Interface      Link   Interface      Link
```

```
pw1                                 ge-11/1/0.0     up     ge-11/1/1.0     up
pw2                                 ge-11/0/0.0     up     ge-11/0/1.0     up
pw3                                 ge-11/1/2.0     down   ge-11/1/3.0     down
Total secure wires: 3
```

## Sample Output

### show security forward-options secure-wire pw1

```
user@host> show security forward-options secure-wire pw1
Secure wire                         Interface       Link   Interface       Link

pw1                                 ge-11/1/0.0     up     ge-11/1/1.0     up
```

## show security policies

**Supported Platforms** SRX Series, vSRX

**Syntax**

**show security policies**
none
<detail>
policy-name *policy-name*
<global>

**Release Information** Command modified in Junos OS Release 9.2. Support for IPv6 addresses added in Junos OS Release 10.2. Support for wildcard addresses added in Junos OS Release 11.1. Support for global policy added in Junos OS Release 11.4. Support for services offloading added in Junos OS Release 11.4. Support for source-identities added in Junos OS Release 12.1. The **Description** output field added in Junos OS Release 12.1. Support for negated address added in Junos OS Release 12.1X45-D10. The output fields for Policy Statistics expanded, and the output fields for the **global** and **policy-name** options expanded to include from-zone and to-zone global match criteria in Junos OS Release 12.1X47-D10. Support for the **initial-tcp-mss** and **reverse-tcp-mss** options added in Junos OS Release 12.3X48-D20. Output field and description for **source-end-user-profile** option added in Junos OS Release 15.1x49-D70. Output field and description for **dynamic-applications** option added in Junos OS Release 15.1x49-D100.

**Description** Display a summary of all security policies configured on the device. If a particular policy is specified, display information specific to that policy.

**Options**

- **none**—Display basic information about all configured policies.
- **detail**—(Optional) Display a detailed view of all of the policies configured on the device.
- **policy-name *policy-name***—(Optional) Display information about a specified policy.
- **global**—(Optional) Display information about global policies.

**Required Privilege Level** view

**Related Documentation**

- *Security Policies Overview*
- *Understanding Security Policy Rules*
- *Understanding Security Policy Elements*

**List of Sample Output**

show security policies detail (TCP Options) on page 527
show security policies policy-name (Negated Address) on page 527
show security policies policy-name detail (Negated Address) on page 528
show security policies global on page 528

**Output Fields** Table 69 on page 521 lists the output fields for the **show security policies** command. Output fields are listed in the approximate order in which they appear.

### Table 69: show security policies Output Fields

| Field Name | Field Description |
|---|---|
| **From zone** | Name of the source zone. |
| **To zone** | Name of the destination zone. |
| **Policy** | Name of the applicable policy. |
| **Description** | Description of the applicable policy. |
| **State** | Status of the policy:<br>• **enabled**: The policy can be used in the policy lookup process, which determines access rights for a packet and the action taken in regard to it.<br>• **disabled**: The policy cannot be used in the policy lookup process, and therefore it is not available for access control. |
| **Index** | Internal number associated with the policy. |
| **Sequence number** | Number of the policy within a given context. For example, three policies that are applicable in a from-zoneA-to-zoneB context might be ordered with sequence numbers 1, 2, 3. Also, in a from-zoneC-to-zoneD context, four policies might have sequence numbers 1, 2, 3, 4. |
| **Source addresses** | For standard display mode, the names of the source addresses for a policy. Address sets are resolved to their individual names.<br>For detail display mode, the names and corresponding IP addresses of the source addresses for a policy. Address sets are resolved to their individual address name-IP address pairs. |
| **Destination addresses** | Name of the destination address (or address set) as it was entered in the destination zone's address book. A packet's destination address must match this value for the policy to apply to it. |
| **source-end-user-profile** | Name of the device identity profile (referred to as **end-user-profile** in the CLI) that contains attributes, or characteristics of a device. Specification of the device identity profile in the **source-end-user-profile** field is part of the device identity feature. If a device matches the attributes specified in the profile and other security policy parameters, then the security policy's action is applied to traffic issuing from the device. |
| **Source addresses (excluded)** | Name of the source address excluded from the policy. |
| **Destination addresses (excluded)** | Name of the destination address excluded from the policy. |

Table 69: show security policies Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| **Source identities** | One or more user roles specified for a policy. |
| **Applications** | Name of a preconfigured or custom application whose type the packet matches, as specified at configuration time.<br>• **IP protocol**: The Internet protocol used by the application—for example, TCP, UDP, ICMP.<br>• **ALG**: If an ALG is explicitly associated with the policy, the name of the ALG is displayed. If **application-protocol ignore** is configured, **ignore** is displayed. Otherwise, **0** is displayed.<br>However, even if this command shows **ALG: 0**, ALGs might be triggered for packets destined to well-known ports on which ALGs are listening, unless ALGs are explicitly disabled or when **application-protocol ignore** is not configured for custom applications.<br>• **Inactivity timeout**: Elapsed time without activity after which the application is terminated.<br>• **Source port range**: The low-high source port range for the session application. |
| **Dynamic Applications** | Application identification based layer 7 dynamic applications. |
| **Destination Address Translation** | Status of the destination address translation traffic:<br>• **drop translated**—Drop the packets with translated destination addresses.<br>• **drop untranslated**—Drop the packets without translated destination addresses. |
| **Application Firewall** | An application firewall includes the following:<br>• **Rule-set**—Name of the rule set.<br>• **Rule**—Name of the rule.<br>◦ **Dynamic applications**—Name of the applications.<br>◦ **Dynamic application groups**—Name of the application groups.<br>◦ **Action**—The action taken with respect to a packet that matches the application firewall rule set. Actions include the following:<br>▪ **permit**<br>▪ **deny**<br>• **Default rule**—The default rule applied when the identified application is not specified in any rules of the rule set. |

Table 69: show security policies Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| **Action or Action-type** | • The action taken in regard to a packet that matches the policy's tuples. Actions include the following:<br>  • **permit**<br>  • **firewall-authentication**<br>  • **tunnel ipsec-vpn** *vpn-name*<br>  • **pair-policy** *pair-policy-name*<br>  • **source-nat pool** *pool-name*<br>  • **pool-set** *pool-set-name*<br>  • **interface**<br>  • **destination-nat** *name*<br>  • **deny**<br>  • **reject**<br>  • **services-offload** |
| **Session log** | Session log entry that indicates whether the **at-create** and **at-close** flags were set at configuration time to log session information. |
| **Scheduler name** | Name of a preconfigured scheduler whose schedule determines when the policy is active and can be used as a possible match for traffic. |
| **Policy statistics** | • **Input bytes**—The total number of bytes presented for processing by the device.<br>  • **Initial direction**—The number of bytes presented for processing by the device from the initial direction.<br>  • **Reply direction**—The number of bytes presented for processing by the device from the reply direction.<br>• **Output bytes**—The total number of bytes actually processed by the device.<br>  • **Initial direction**—The number of bytes from the initial direction actually processed by the device.<br>  • **Reply direction**—The number of bytes from the reply direction actually processed by the device.<br>• **Input packets**—The total number of packets presented for processing by the device.<br>  • **Initial direction**—The number of packets presented for processing by the device from the initial direction.<br>  • **Reply direction**—The number of packets presented for processing by the device from the reply direction.<br>• **Output packets**—The total number of packets actually processed by the device.<br>  • **Initial direction**—The number of packets actually processed by the device from the initial direction.<br>  • **Reply direction**—The number of packets actually processed by the device from the reply direction.<br>• **Session rate**—The total number of active and deleted sessions.<br>• **Active sessions**—The number of sessions currently present because of access control lookups that used this policy.<br>• **Session deletions**—The number of sessions deleted since system startup.<br>• **Policy lookups**—The number of times the policy was accessed to check for a match. |

Table 69: show security policies Output Fields *(continued)*

| Field Name | Field Description |
|---|---|
| **Per policy TCP Options** | Configured syn and sequence checks, and the configured TCP MSS value for the initial direction and /or the reverse direction. |

## Sample Output

### show security policies

```
user@host> show security policies
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Sequence number: 1
    Source addresses:
    sa-1-ipv4: 198.51.100.11/24
    sa-2-ipv6: 2001:db8:a0b:12f0::1/32
    sa-3-ipv6: 2001:db8:a0b:12f0::22/32
    sa-4-wc:   203.0.113.1/255.255.0.255
    Destination addresses:
    da-1-ipv4: 2.2.2.2/24
    da-2-ipv6: 2001:db8:a0b:12f0::8/32
    da-3-ipv6: 2001:db8:a0b:12f0::9/32
    da-4-wc:   192.168.22.11/255.255.0.255
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, application services, log, scheduled
    Application firewall : my_ruleset1
  Policy: p2, State: enabled, Index: 5, Sequence number: 2
    Source addresses:
    sa-1-ipv4:  198.51.100.11/24
    sa-2-ipv6:  2001:db8:a0b:12f0::1/32
    sa-3-ipv6:  2001:db8:a0b:12f0::22/32
    Destination addresses:
    da-1-ipv4:  2.2.2.2/24
    da-2-ipv6:  2001:db8:a0b:12f0::1/32
    da-3-ipv6:  2001:db8:a0b:12f0::9/32
    Source identities: role1, role4
    Applications: any
    Action: deny, scheduled
```

### show security policies (Dynamic Applications)

```
user@host> show security policies
Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Applications: any
    Dynamic Applications: junos:YAHOO
    Action: deny, log
  Policy: p2, State: enabled, Index: 5, Scope Policy: 0, Sequence number: 2
    Source addresses: any
    Destination addresses: any
    Applications: any
    Dynamic Applications: junos:web, junos:web:social-networking:facebook,
    junos:TFTP, junos:QQ
    Action: permit, log
  Policy: p3, State: enabled, Index: 6, Scope Policy: 0, Sequence number: 3
    Source addresses: any
```

```
    Destination addresses: any
    Applications: any
    Dynamic Applications: junos:HTTP, junos:SSL
    Action: permit, application services, log
```

## show security policies policy-name detail

```
user@host> show security policies policy-name p1 detail
Policy: p1, action-type: permit, State: enabled, Index: 4
  Description: The policy p1 is for the sales team
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses:
    sa-1-ipv4:  198.51.100.11/24
    sa-2-ipv6:  2001:db8:a0b:12f0::1/32
    sa-3-ipv6:  2001:db8:a0b:12f0::9/32
    sa-4-wc:    203.0.113.1/255.255.0.255
  Destination addresses:
    da-1-ipv4: 192.0.2.0/24
    da-2-ipv6: 2001:db8:a0b:12f0::1/32
    da-3-ipv6: 2001:db8:a0b:12f0::9/32
    da-4-wc:   192.168.22.11/255.255.0.255
 Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Destination Address Translation: drop translated
  Application firewall :
  Rule-set: my_ruleset1
    Rule: rule1
      Dynamic Applications: junos:FACEBOOK-ACCESS, junos:YMSG
      Dynamic Application groups: junos:web, junos:chat
      Action: deny
    Default rule: permit
  Session log: at-create, at-close
  Scheduler name: sch20
  Per policy TCP Options: SYN check: No, SEQ check: No
  Policy statistics:
    Input  bytes       :               18144                 545 bps
      Initial direction:                9072                272 bps
      Reply direction  :                9072                272 bps
    Output bytes       :               18144                 545 bps
      Initial direction:                9072                272 bps
      Reply direction  :                9072                272 bps
    Input  packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
    Output packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
    Session rate       :                 108                   3 sps
    Active sessions    :                  93
    Session deletions  :                  15
    Policy lookups     :                 108
```

### show security policies (Services-Offload)

```
user@host> show security policies
Default policy: deny-all
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, services-offload, count
From zone: untrust, To zone: trust
  Policy: p2, State: enabled, Index: 5, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, services-offload
```

### show security policies (Device Identity)

```
user@host>  show security policies
From zone: trust, To zone: untrust
       Policy: dev-id-marketing, State: enabled, Index: 5, Scope Policy: 0,
Sequence number: 1
         Source addresses: any
         Destination addresses: any
         source-end-user-profile: marketing-profile
         Applications: any
         Action: permit
```

### show security policies detail

```
user@host> show security policies detail
Default policy: deny-all
Policy: p1, action-type: permit, services-offload:enabled , State: enabled, Index:
 4, Scope Policy: 0
  Policy Type: Configured
  Description: The policy p1 is for the sales team
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
 Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
  Policy statistics:
    Input  bytes       :                18144                 545 bps
      Initial direction:                 9072                272 bps
      Reply direction  :                 9072                 272 bps
```

```
    Output bytes       :                18144                  545 bps
      Initial direction:                 9072                  272 bps
      Reply direction  :                 9072                  272 bps
    Input  packets     :                  216                    6 pps
      Initial direction:                  108                    3 bps
      Reply direction  :                  108                    3 bps
    Output packets     :                  216                    6 pps
      Initial direction:                  108                    3 bps
      Reply direction  :                  108                    3 bps
    Session rate       :                  108                    3 sps
    Active sessions    :                   93
    Session deletions  :                   15
    Policy lookups     :                  108
Policy: p2, action-type: permit, services-offload:enabled , State: enabled, Index:
 5, Scope Policy: 0
  Policy Type: Configured
  Description: The policy p2 is for the sales team
  Sequence number: 1
  From zone: untrust, To zone: trust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
 Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
```

## show security policies detail (TCP Options)

```
user@host> show security policies policy-name policy1 detail
node0:
--------------------------------------------------------------------------
Policy: policy1, action-type: permit, State: enabled, Index: 7, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 2
  From zone: trust, To zone: untrust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
  Per policy TCP MSS: initial: 800, reverse: 900
```

## show security policies policy-name (Negated Address)

```
user@host>  show security policies policy-name p1
```

```
node0:
--------------------------------------------------------------------------
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses(excluded): as1
    Destination addresses(excluded): as2
    Applications: any
    Action: permit
```

### show security policies policy-name detail (Negated Address)

```
user@host> show security policies policy-name p1 detail
node0:
--------------------------------------------------------------------------
Policy: p1, action-type: permit, State: enabled, Index: 4, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses(excluded):
    ad1(ad): 255.255.255.255/32
    ad2(ad):  198.51.100.1/24
    ad3(ad):  198.51.100.6  ~ 198.51.100.56
    ad4(ad):  192.0.2.8/24
    ad5(ad):  198.51.100.99 ~ 198.51.100.199
    ad6(ad):  203.0.113.9/24
    ad7(ad):  203.0.113.23/24
  Destination addresses(excluded):
    ad13(ad2): 198.51.100.76/24
    ad12(ad2): 198.51.100.88/24
    ad11(ad2): 192.0.2.23 ~ 192.0.2.66
    ad10(ad2): 192.0.2.93
    ad9(ad2):  203.0.113.76 ~ 203.0.113.106
    ad8(ad2):  203.0.113.199
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
```

### show security policies global

```
user@host> show security policies global policy-name Pa
node0:
--------------------------------------------------------------------------
  Global policies:
  Policy: Pa, State: enabled, Index: 5, Scope Policy: 0, Sequence number: 1
    From zones: zone1, zone2
    To zones: zone3, zone4    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

## show security zones

**Supported Platforms** SRX Series, vSRX

**Syntax** **show security zones** <*zone-name*> <**detail** | **terse**>

**Release Information** Command introduced in Junos OS Release 8.5. The **Description** output field added in Junos OS Release 12.1.

**Description** Display information about security zones.

**Options**
- **none**—Display information about all zones.
- **detail** | **terse**—(Optional) Display the specified level of output.
- ***zone-name***—(Optional) Display information about the specified zone.

**Required Privilege Level** view

**Related Documentation**
- *Security Zones and Interfaces Overview*
- *Supported System Services for Host Inbound Traffic*
- security-zone on page 369

**List of Sample Output**
show security zones on page 530
show security zones abc on page 531
show security zones abc detail on page 531
show security zones terse on page 531

**Output Fields** Table 70 on page 529 lists the output fields for the **show security zones** command. Output fields are listed in the approximate order in which they appear.

### Table 70: show security zones Output Fields

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Functional zone** | Name of the functional zone. | none |
| **Security zone** | Name of the security zone. | detail<br>none |
| **Description** | Description of the security zone. | detail<br>none |

Table 70: show security zones Output Fields *(continued)*

| Field Name | Field Description | Level of Output |
|---|---|---|
| **Policy configurable** | Whether the policy can be configured or not. | detail<br>none |
| **Interfaces bound** | Number of interfaces in the zone. | detail<br>none |
| **Interfaces** | List of the interfaces in the zone. | detail<br>none |
| **Zone** | Name of the zone. | terse |
| **Type** | Type of the zone. | terse |

## Sample Output

### show security zones

```
user@host> show security zones
Functional zone: management
   Description: This is the management zone.
   Policy configurable: No
   Interfaces bound: 1
   Interfaces:
     ge-0/0/0.0
Security zone: Host
   Description: This is the host zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     fxp0.0
Security zone: abc
   Description: This is the abc zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/1.0
Security zone: def
   Description: This is the def zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/2.0
```

## Sample Output

### show security zones abc

```
user@host> show security zones abc
Security zone: abc
   Description: This is the abc zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/1.0
```

## Sample Output

### show security zones abc detail

```
user@host> show security zones abc detail
Security zone: abc
   Description: This is the abc zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/1.0
```

## Sample Output

### show security zones terse

```
user@host> show security zones terse
Zone                    Type
my-internal             Security
my-external             Security
dmz                     Security
```

## show spanning-tree interface

**Supported Platforms** SRX Series

**Syntax**
show spanning-tree interface
<brief | detail>
<interface-name *interface-name*>
<msti *msti-id*>
<routing-instances *routing-instance-name*>
<vlan-id *vlan-id*>

**Release Information** Command introduced in Junos OS Release 11.1.

**Description** Display the configured or calculated interface-level spanning-tree protocol (can be either STP, RSTP, or MSTP) parameters. In **brief** mode, the command output does not display interfaces that are administratively disabled or do not have a physical link.

**Options**
**brief | detail**—(Optional) Display the specified level of output.

**interface-name *interface-name***—(Optional) Name of an interface.

**msti *msti-id***—(Optional) Display STP bridge information for the specified MSTP instance ID or common and internal spanning tree (CIST). Specify **0** for CIST. Specify a value from **1** through **64** for an MSTI.

**vlan-id *vlan-id***—(Optional) For MSTP interfaces, display interface information for the specified VLAN. Specify a value from **0** through **4094**.

**Required Privilege Level** view

**Related Documentation**
- Understanding the Spanning Tree Protocol on page 147
- Configuring the Spanning Tree Protocol (J- Web Procedure) on page 151
- Configuring the Spanning Tree Protocol (CLI Procedure) on page 152

**List of Sample Output**
show spanning-tree interface on page 533
show spanning-tree interface brief on page 534
show spanning-tree interface detail on page 534
show spanning-tree interface (Specified Interface) on page 535

**Output Fields** Table 71 on page 533 lists the output fields for the **show spanning-tree Interface** command. Output fields are listed in the approximate order in which they appear.

Table 71: show spanning-tree interface Output Fields

| Field Name | Field Description |
|---|---|
| **Interface name** | Interface configured to participate in the STP, RSTP, or MSTP instance. |
| **Port ID** | Logical interface identifier configured to participate in the MSTP instance. |
| **Designated port ID** | Port ID of the designated port for the LAN segment this interface is attached to. |
| **Designated bridge ID** | Bridge ID of the designated bridge for the LAN segment this interface is attached to. |
| **Port Cost** | Configured cost for the interface. |
| **State** | STP port state. Forwarding (FWD), blocking (BLK), listening, learning, or disabled. |
| **Role** | MSTP or RSTP port role. Designated (DESG), backup (BKUP), alternate (ALT), (ROOT), or Root Prevented (Root-Prev). |
| **Link type** | MSTP or RSTP link type. Shared or point-to-point (pt-pt) and edge or non edge. |
| **Alternate** | Identifies the interface as an MSTP or RSTP alternate root port (yes) or non-alternate root port (no). |
| **Boundary Port** | Identifies the interface as an MSTP regional boundary port (yes) or non-boundary port (no). |
| **Edge delay while expiry count** | Number of times the edge delay timer expired on that interface. |
| **Rcvd info while expiry count** | Number of times the rcvd info timer expired on that interface. |

## Sample Output

### show spanning-tree interface

```
user@host> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated  Designated     Port    State  Role
                          port ID    bridge ID      Cost
ge-0/0/0.0  128:513  128:513   8192.0019e2500340    1000   FWD    DESG
ge-0/0/2.0  128:515  128:515   8192.0019e2500340    1000   BLK    DIS
ge-0/0/4.0  128:517  128:517   8192.0019e2500340    1000   FWD    DESG
ge-0/0/23.0 128:536  128:536   8192.0019e2500340    1000   FWD    DESG


Spanning tree interface parameters for instance 1

Interface    Port ID    Designated  Designated     Port    State  Role
                          port ID    bridge ID      Cost
ge-0/0/0.0  128:513  128:513   8193.0019e2500340    1000   FWD    DESG
ge-0/0/2.0  128:515  128:515   8193.0019e2500340    1000   BLK    DIS
```

```
ge-0/0/4.0  128:517  128:517   8193.0019e2500340    1000   FWD    DESG
ge-0/0/23.0 128:536  128:536   8193.0019e2500340    1000   FWD    DESG

Spanning tree interface parameters for instance 2

Interface    Port ID    Designated  Designated     Port    State  Role
                         port ID    bridge ID      Cost
ge-0/0/0.0  128:513  128:1   8194.001b549fd000  1000  FWD    ROOT
ge-0/0/2.0  128:515  128:515  32770.0019e2500340  4000  BLK    DIS
ge-0/0/4.0  128:517  128:1    16386.001b54013080  1000  BLK    ALT
ge-0/0/23.0 128:536  128:536  32770.0019e2500340  1000  FWD    DESG
```

## show spanning-tree interface brief

```
user@host> show spanning-tree interface brief
Spanning tree interface parameters for instance 0

Interface    Port ID    Designated  Designated     Port    State  Role
     port ID    bridge ID  Cost
ge-1/0/0.0  128:625  128:625  32768.0019e25095a0 20000  BLK    DIS
ge-1/0/1.0  128:626  128:626  32768.0019e25095a0 20000  BLK    DIS
ge-1/0/2.0  128:627  128:627  32768.0019e25095a0 20000  BLK    DIS
ge-1/0/10.0 128:635  128:635  32768.0019e25095a0 20000  BLK    DIS
ge-1/0/20.0 128:645  128:645  32768.0019e25095a0 20000  BLK    DIS
ge-1/0/30.0 128:655  128:655  32768.0019e25095a0 20000  BLK    DIS
```

## show spanning-tree interface detail

```
user@host> show spanning-tree interface detail
Spanning tree interface parameters for instance 0

Interface name     : ge-1/0/0.0
Port identifier    : 128.625
Designated port ID     : 128.625
Port cost      : 20000
Port state     : Blocking
Designated bridge ID   : 32768.00:19:e2:50:95:a0
Port role      : Disabled
Link type      : Pt-Pt/EDGE
Boundary port       : NA
Edge delay while expiry count  : 0
Rcvd info while expiry count   : 0

Interface name     : ge-1/0/1.0
Port identifier    : 128.626
Designated port ID     : 128.626
Port cost      : 20000
Port state     : Blocking
Designated bridge ID   : 32768.00:19:e2:50:95:a0
Port role      : Disabled
Link type      : Pt-Pt/NONEDGE
Boundary port       : NA
Edge delay while expiry count  : 0
Rvcd info while expiry count   : 0

Interface name     : ge-1/0/2.0
Port identifier    : 128.627
Designated port ID     : 128.627
Port cost      : 20000
```

```
Port state     : Blocking
Designated bridge ID   : 32768.00:19:e2:50:95:a0
Port role      : Disabled
Link type      : Pt-Pt/NONEDGE
Boundary port       : NA
Edge delay while expiry count  : 0
Rvcd info while expiry count   : 0

Interface name     : ge-1/0/10.0
Port identifier    : 128.635
Designated port ID     : 128.635
Port cost      : 20000
Port state     : Blocking
Designated bridge ID   : 32768.00:19:e2:50:95:a0
Port role      : Disabled
Link type      : Pt-Pt/NONEDGE
Boundary port       : NA
Edge delay while expiry count  : 0
Rvcd info while expiry count   : 0

Interface name     : ge-1/0/20.0
Port identifier    : 128.645
Designated port ID     : 128.645
Port cost      : 20000
Port state     : Blocking
Designated bridge ID   : 32768.00:19:e2:50:95:a0
Port role      : Disabled
Link type      : Pt-Pt/NONEDGE
Boundary port       : NA
Edge delay while expiry count  : 0
Rvcd info while expiry count   : 0
[output truncated]
```

## show spanning-tree interface (Specified Interface)

```
user@host> show spanning-tree interface ge-1/0/0

Interface    Port ID    Designated  Designated     Port    State  Role
     port ID    bridge ID  Cost
ge-1/0/0.0 128:625  128:625  32768.0019e25095a0 20000  BLK    DIS
```

## show vlans

**Supported Platforms** SRX Series, vSRX

**Syntax**
show vlans
<brief | detail | extensive>
<interface *interface-name*>
<logical-system (*logical-system* | *all*)>
<operational>

**Release Information** Command introduced in Junos OS Release 8.4.

**Description** Display VLAN information.

**Options**
**none**—Display information for all VLANs.

**brief | detail | extensive**—(Optional) Display the specified level of output.

**interface *interface-name***— (Optional) Display information about a specific interface.

**logical system**—(Optional) Display name of the logical system or all.

**operational**—(Optional) Display information for the operational switching instances.

**Required Privilege Level** view

**Related Documentation**
- show ethernet-switching mac-learning-log (View) on page 419
- show ethernet-switching table (View) on page 421

**List of Sample Output**


### Sample Output

#### show vlans

```
user@host> show vlans
Routing instance        VLAN name             Tag          Interfaces
default-switch          vlan-22                 22
default-switch          vlan-333                333
                                                             ge-0/0/3.0*
                                                             ge-0/0/4.0*
default-switch          default               1
default-switch          vlan100               100
                                                             ge-0/0/1.0*
```

## show vlans brief

```
user@host> show vlans brief
Routing instance        VLAN name             Tag          Interfaces
default-switch          vlan-22                 22
                                                           ge-0/0/3.0*
                                                           ge-0/0/4.0*
default-switch          vlan-333                333
default-switch          default               1
                                                           ge-0/0/1.0*
default-switch          vlan100               100
```

## show vlans detail

```
user@host> show vlans detail
Routing instance: default-switch
  VLAN Name: vlan-22                            State: Active
Tag: 22
Internal index: 2, Generation Index: 1, Origin: Static
MAC aging time: 300 seconds
VXLAN Enabled : No
Number of interfaces: Tagged 0    , Untagged 0
Total MAC count: 0

Routing instance: default-switch
  VLAN Name: vlan-333                           State: Active
Tag: 333
Internal index: 3, Generation Index: 2, Origin: Static
MAC aging time: 300 seconds
VXLAN Enabled : No
Interfaces:
    ge-0/0/3.0*,tagged,trunk
    ge-0/0/4.0*,tagged,trunk
Number of interfaces: Tagged 2    , Untagged 0
Total MAC count: 0
```