

**Junos<sup>®</sup> OS**

---

# Layer 2 VPNs User Guide for EX9200 Switches

Published  
2020-03-12

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

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

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

*Junos<sup>®</sup> OS Layer 2 VPNs User Guide for EX9200 Switches*  
Copyright © 2020 Juniper Networks, Inc. All rights reserved.

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

## YEAR 2000 NOTICE

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

## END USER LICENSE AGREEMENT

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

# Table of Contents

## About the Documentation | x

Documentation and Release Notes | x

Using the Examples in This Manual | x

    Merging a Full Example | xi

    Merging a Snippet | xi

Documentation Conventions | xii

Documentation Feedback | xv

Requesting Technical Support | xv

    Self-Help Online Tools and Resources | xvi

    Creating a Service Request with JTAC | xvi

## 1

## Common Configuration for Layer 2 VPNs

### Overview | 2

Understanding Layer 2 VPNs | 2

Layer 2 VPN Applications | 3

Supported Layer 2 VPN Standards | 4

### Pinging VPNs | 6

Pinging VPNs, VPLS, and Layer 2 Circuits | 6

Pinging a Layer 2 VPN | 7

Pinging a Layer 2 Circuit | 7

### Layer 2 VPNs Configuration Overview | 8

Introduction to Configuring Layer 2 VPNs | 8

Configuring the Local Site on PE Routers in Layer 2 VPNs | 10

    Configuring a Layer 2 VPN Routing Instance | 10

    Configuring the Site | 11

    Configuring the Remote Site ID | 12

    Configuring the Encapsulation Type | 13

    Configuring a Site Preference and Layer 2 VPN Multihoming | 14

Tracing Layer 2 VPN Traffic and Operations | 15

Disabling Normal TTL Decrementing for VPNs | 16

Example: Configuring MPLS-Based Layer 2 VPNs | 16

## **Configuring Layer 2 Interfaces | 34**

Configuring CCC Encapsulation for Layer 2 VPNs | 34

Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits | 35

Configuring the MTU for Layer 2 Interfaces | 37

Disabling the Control Word for Layer 2 VPNs | 38

## **Configuring Path Selection for Layer 2 VPNs and VPLS | 40**

Understanding BGP Path Selection | 40

Routing Table Path Selection | 42

BGP Table path selection | 44

Effects of Advertising Multiple Paths to a Destination | 45

Enabling BGP Path Selection for Layer 2 VPNs and VPLS | 46

## **Creating Backup Connections with Redundant Pseudowires | 49**

Redundant Pseudowires for Layer 2 Circuits and VPLS | 49

Types of Redundant Pseudowire Configurations | 50

Pseudowire Failure Detection | 51

Configuring Redundant Pseudowires for Layer 2 Circuits and VPLS | 52

Configuring Pseudowire Redundancy on the PE Router | 52

Configuring the Switchover Delay for the Pseudowires | 53

Configuring a Revert Time for the Redundant Pseudowire | 53

## **Monitoring Layer 2 VPNs Using BFD | 55**

Configuring BFD for Layer 2 VPN and VPLS | 56

BFD Support for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS | 58

Configuring BFD for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS | 59

## Configuring Layer 2 Circuits

### Overview | 62

Layer 2 Circuit Overview | 62

### Layer 2 Circuits Configuration Overview | 64

Configuring Static Layer 2 Circuits | 64

Configuring Local Interface Switching in Layer 2 Circuits | 65

Configuring the Interfaces for the Local Interface Switch | 66

Enabling Local Interface Switching When the MTU Does Not Match | 67

Configuring Interfaces for Layer 2 Circuits | 68

Configuring the Address for the Neighbor of the Layer 2 Circuit | 68

Configuring the Neighbor Interface for the Layer 2 Circuit | 69

Configuring a Community for the Layer 2 Circuit | 70

Configuring the Control Word for Layer 2 Circuits | 70

Configuring the Encapsulation Type for the Layer 2 Circuit Neighbor Interface | 72

Enabling the Layer 2 Circuit When the Encapsulation Does Not Match | 72

Configuring the MTU Advertised for a Layer 2 Circuit | 73

Enabling the Layer 2 Circuit When the MTU Does Not Match | 73

Configuring the Protect Interface | 73

Configuring the Protect Interface From Switching Over to the Primary Interface | 74

Configuring the Pseudowire Status TLV | 74

Configuring Layer 2 Circuits over Both RSVP and LDP LSPs | 75

Configuring the Virtual Circuit ID | 76

Configuring the Interface Encapsulation Type for Layer 2 Circuits | 76

Configuring ATM2 IQ Interfaces for Layer 2 Circuits | 77

Configuring Policies for Layer 2 Circuits | 78

Configuring the Layer 2 Circuit Community | 78

Configuring the Policy Statement for the Layer 2 Circuit Community | 79

Example: Configuring a Policy for a Layer 2 Circuit Community | 80

Verifying the Layer 2 Circuit Policy Configuration | 81

Configuring LDP for Layer 2 Circuits | 81

## **Configuring Protection Features for Layer 2 Circuits | 83**

Egress Protection LSPs for Layer 2 Circuits | 83

Example: Configuring Layer 2 Circuit Switching Protection | 84

## **Monitoring Layer 2 Circuits with BFD | 102**

Configuring BFD for VCCV for Layer 2 Circuits | 102

Example: Configuring BFD for VCCV for Layer 2 Circuits | 105

## **Troubleshooting Layer 2 Circuits | 116**

Tracing Layer 2 Circuit Operations | 116

## **Configuration Statements and Operational Commands**

### **Configuration Statements (All VPNs) | 118**

aggregate-label | 119

backup-neighbor | 120

description (Routing Instances) | 122

family route-target | 123

graceful-restart (Enabling Globally) | 125

instance-type | 127

interface (Routing Instances) | 130

no-forwarding | 131

forward-policy-mismatch (Security Group VPN Member) | 132

proxy-generate | 133

revert-time (Protocols Layer 2 Circuits) | 134

route-distinguisher | 136

route-distinguisher-id | 140

route-target-filter | 141

switchover-delay | 143

unicast-reverse-path | 144

vpn-apply-export | 145

vrf-export | 146

vrf-import | 148

vrf-mtu-check | 149

vrf-target | 150

## **Configuration Statements (Layer 2 VPNs) | 152**

auto-discovery-only | 155

backup-interface (Layer 2 Circuits) | 156

bfd-liveness-detection (Layer 2 VPN and VPLS) | 157

community (Protocols Layer 2 Circuit) | 159

connection-protection | 160

control-channel (Protocols OAM) | 161

control-word (Protocols Layer 2 Circuit Neighbor) | 162

control-word (Protocols Layer 2 VPN) | 163

description (Protocols Layer 2 Circuit Neighbor) | 164

description (Protocols Layer 2 VPN) | 165

detection-time (BFD Liveness Detection) | 166

egress-protection (Layer 2 circuit) | 169

egress-protection (MPLS) | 170

encapsulation (Logical Interface) | 172

encapsulation | 177

encapsulation-type (Layer 2 Circuits) | 184

encapsulation-type (Layer 2 VPNs) | 186

end-interface | 188

family (Protocols BGP) | 189

family multiservice | 195

flow-label-receive-static | 198

flow-label-transmit-static | 199

hot-standby | 200

hot-standby (Protocols Layer 2 Circuit) | 201

hot-standby-vc-on (Protocols Layer 2 Circuit) | 202

ignore-encapsulation-mismatch | 204

ignore-mtu-mismatch | 205

interface (Protocols Layer 2 Circuit) | 206

interface (Protocols Layer 2 VPN) | 208

install-nexthop | 209

l2circuit | 210

l2ckt | 212

l2vpn | 213

l2vpn (routing-options) | 216

l2vpn-id | 217

local-switching (Layer 2 Circuits) | 218

minimum-interval (BFD Liveness Detection) | 219

minimum-receive-interval (BFD Liveness Detection) | 221

mtu | 223

multiplier (BFD Liveness Detection) | 227

neighbor (Protocols Layer 2 Circuit) | 229

no-adaptation (BFD Liveness Detection) | 231

no-control-word (Protocols Layer 2 VPN) | 233

no-l2ckt | 234

no-l2vpn | 235

no-revert (Local Switching) | 236

no-revert (Neighbor Interface) | 237

oam | 238

path-selection | 240

ping-interval | 243

policer (Layer 2 VPN) | 244

protect-interface | 245

protected-l2circuit | 246

protector-interface | 247

protector-pe | 248

pseudowire-status-tlv | 249

psn-tunnel-endpoint | 250

remote-site-id | 251

routing-instances | 252

send-oam | 253

site (Layer 2 Circuits) | 254

site-identifier (Layer 2 Circuits) | 255

site-preference | 256

source-attachment-identifier (Protocols VPWS) | 257

standby (Protocols Layer 2 Circuit) | 259



static (Protocols Layer 2 Circuit) | 260

target-attachment-identifier (Protocols VPWS) | 262

template | 263

traceoptions (Egress Protection) | 264

traceoptions (Protocols Layer 2 Circuit) | 266

traceoptions (Protocols Layer 2 VPN) | 268

transmit-interval (BFD Liveness Detection) | 270

version (BFD Liveness Detection) | 273

virtual-circuit-id | 275

vlan-id | 276

vlan-id (routing instance) | 277

vlan-tagging | 278

## **Operational Commands | 281**

clear pim snooping join | 282

clear pim snooping statistics | 284

ping mpls l2circuit | 287

ping mpls l2vpn | 290

request l2circuit-switchover | 293

show interfaces lsi (Label-Switched Interface) | 295

show l2circuit connections | 299

show l2vpn connections | 310

show pim snooping interfaces | 319

show pim snooping join | 323

show pim snooping neighbors | 328

show pim snooping statistics | 335

show route | 341

# About the Documentation

## IN THIS SECTION

- Documentation and Release Notes | x
- Using the Examples in This Manual | x
- Documentation Conventions | xii
- Documentation Feedback | xv
- Requesting Technical Support | xv

## 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>.

## 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.xml; }
```

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

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

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

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

For more information about the **load** command, see [CLI Explorer](#).

## Documentation Conventions

[Table 1 on page xiii](#) defines notice icons used in this guide.

Table 1: Notice Icons






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

Table 2 on page [xiii](#) defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
<b>Bold text like this</b>	Represents text that you type.	To enter configuration mode, type the <b>configure</b> command:  user@host> <b>configure</b>
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> <b>show chassis alarms</b>  No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> <li>Introduces or emphasizes important new terms.</li> <li>Identifies guide names.</li> <li>Identifies RFC and Internet draft titles.</li> </ul>	<ul style="list-style-type: none"> <li>A policy <i>term</i> is a named structure that defines match conditions and actions.</li> <li><i>Junos OS CLI User Guide</i></li> <li>RFC 1997, <i>BGP Communities Attribute</i></li> </ul>

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

Convention	Description	Examples
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name:  [edit] root@# <b>set system domain-name</b> <i>domain-name</i>
<b>Text like this</b>	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"><li>• To configure a stub area, include the <b>stub</b> statement at the [edit <b>protocols ospf area area-id</b>] hierarchy level.</li><li>• The console port is labeled <b>CONSOLE</b>.</li></ul>
< > (angle brackets)	Encloses optional keywords or variables.	<b>stub &lt;default-metric <i>metric</i>&gt;;</b>
(pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	<b>broadcast   multicast</b>  <b>(<i>string1</i>   <i>string2</i>   <i>string3</i>)</b>
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	<b>rsvp { # Required for dynamic MPLS only</b>
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	<b>community name members [ <i>community-ids</i> ]</b>
Indentation and braces ( { } )	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		

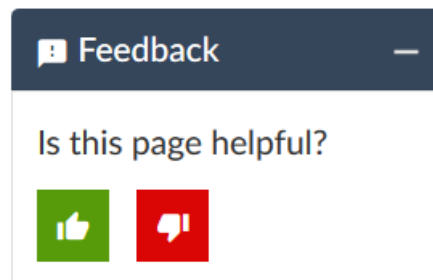
Table 2: Text and Syntax Conventions (*continued*)

Convention	Description	Examples
<b>Bold text like this</b>	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none"> <li>In the Logical Interfaces box, select <b>All Interfaces</b>.</li> <li>To cancel the configuration, click <b>Cancel</b>.</li> </ul>
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select <b>Protocols&gt;Ospf</b> .

## Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the [Juniper Networks TechLibrary](#) site, and do one of the following:



- Click the thumbs-up icon if the information on the page was helpful to you.
- Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.
- E-mail—Send your comments to [techpubs-comments@juniper.net](mailto:techpubs-comments@juniper.net). Include the document or topic name, URL or page number, and software version (if applicable).

## Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services 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/>
- Create a service request online: <https://myjuniper.juniper.net>

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

## Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit <https://myjuniper.juniper.net>.
- 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://support.juniper.net/support/requesting-support/>.



# 1

PART

## Common Configuration for Layer 2 VPNs

---

[Overview | 2](#)

[Pinging VPNs | 6](#)

[Layer 2 VPNs Configuration Overview | 8](#)

[Configuring Layer 2 Interfaces | 34](#)

[Configuring Path Selection for Layer 2 VPNs and VPLS | 40](#)

[Creating Backup Connections with Redundant Pseudowires | 49](#)

[Monitoring Layer 2 VPNs Using BFD | 55](#)

---

# Overview

## IN THIS CHAPTER

- Understanding Layer 2 VPNs | 2
- Layer 2 VPN Applications | 3
- Supported Layer 2 VPN Standards | 4

## Understanding Layer 2 VPNs

**NOTE:** On EX9200 switches, graceful Routing Engine switchover (GRES), nonstop active routing (NSR), and logical systems are not supported on Layer 2 VPN configurations. Layer 2 VPN is not supported on the EX9200 Virtual Chassis.

As the need to link different Layer 2 services to one another for expanded service offerings grows, Layer 2 Multiprotocol Label Switching (MPLS) VPN services are increasingly in demand.

Implementing a Layer 2 VPN on a router is similar to implementing a VPN using a Layer 2 technology such as Asynchronous Transfer Mode (ATM) or Frame Relay. However, for a Layer 2 VPN on a router, traffic is forwarded to the router in a Layer 2 format. It is carried by MPLS over the service provider's network, and then converted back to Layer 2 format at the receiving site. You can configure different Layer 2 formats at the sending and receiving sites. The security and privacy of an MPLS Layer 2 VPN are equal to those of an ATM or Frame Relay VPN. The service provisioned with Layer 2 VPNs is also known as Virtual Private Wire Service (VPWS).

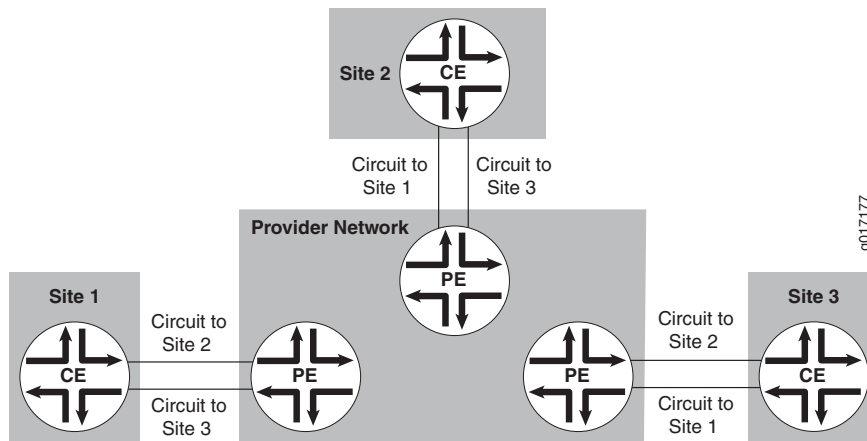
On a Layer 2 VPN, routing occurs on the customer's routers, typically on the customer edge (CE) router. The CE router connected to a service provider on a Layer 2 VPN must select the appropriate circuit on which to send traffic. The provider edge (PE) router receiving the traffic sends it across the service provider's network to the PE router connected to the receiving site. The PE routers do not need to store or process the customer's routes; they only need to be configured to send data to the appropriate tunnel.

For a Layer 2 VPN, customers need to configure their own routers to carry all Layer 3 traffic. The service provider needs to know only how much traffic the Layer 2 VPN will need to carry. The service provider's

routers carry traffic between the customer's sites using Layer 2 VPN interfaces. The VPN topology is determined by policies configured on the PE routers.

Customers need to know only which VPN interfaces connect to which of their own sites. [Figure 1 on page 3](#) illustrates a Layer 2 VPN in which each site has a VPN interface linked to each of the other customer sites.

**Figure 1: Layer 2 VPN Connecting CE Routers**



Implementing a Layer 2 MPLS VPN includes the following benefits:

- Service providers do not have to invest in separate Layer 2 equipment to provide Layer 2 VPN service. A Layer 2 MPLS VPN allows you to provide Layer 2 VPN service over an existing IP and MPLS backbone.
- You can configure the PE router to run any Layer 3 protocol in addition to the Layer 2 protocols.
- Customers who prefer to maintain control over most of the administration of their own networks might want Layer 2 VPN connections with their service provider instead of a Layer 3 VPN.
- Because Layer 2 VPNs use BGP as the signaling protocol, they have a simpler design and require less overhead than traditional VPNs over Layer 2 circuits. BGP signaling also enables autodiscovery of Layer 2 VPN peers. Layer 2 VPNs are similar to BGP or MPLS VPNs and VPLS in many respects; all three types of services employ BGP for signaling.

## Layer 2 VPN Applications

Implementing a Layer 2 VPN includes the following benefits:

- Terminating a Layer 2 VPN into a Layer 2 VPN using the interworking (iw0) software interface eliminates the limitation of bandwidth on the tunnel interfaces used for these configuration scenarios. Instead of using a physical Tunnel PIC for looping the packet received from the Layer 2 VPN to another Layer 2 VPN, Junos OS is used to link both the Layer 2 VPN routes.
- Layer 2 VPNs enable the sharing of a provider's core network infrastructure between IP and Layer 2 VPN services, reducing the cost of providing those services. A Layer 2 MPLS VPN allows you to provide Layer 2 VPN service over an existing IP and MPLS backbone.
- From a service provider's point of view, a Layer 2 MPLS VPN allows the use of a single Layer 3 VPN (such as RFC 2547bis), MPLS traffic engineering, and Differentiated Services (DiffServ).
- Service providers do not have to invest in separate Layer 2 equipment to provide Layer 2 VPN service. You can configure the PE router to run any Layer 3 protocol in addition to the Layer 2 protocols. Customers who prefer to maintain control over most of the administration of their own networks might want Layer 2 VPN connections with their service provider instead of a Layer 3 VPN.

## RELATED DOCUMENTATION

### [Understanding Layer 2 VPNs | 2](#)

*Using the Layer 2 Interworking Interface to Interconnect a Layer 2 Circuit to a Layer 2 VPN*

*Using the Layer 2 Interworking Interface to Interconnect a Layer 2 VPN to a Layer 2 VPN*

*Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN*

*Example: Interconnecting a Layer 2 VPN with a Layer 2 VPN*

*Example: Interconnecting a Layer 2 VPN with a Layer 3 VPN*

## Supported Layer 2 VPN Standards

Junos OS substantially supports the following standards and Internet drafts, which define standards for Layer 2 virtual private networks (VPNs).

- RFC 7348, *Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks*
- Internet draft draft-kompella-l2vpn-vpls-multihoming, *Multi-homing in BGP-based Virtual Private LAN Service*
- Internet draft draft-kompella-ppvpn-l2vpn-03.txt, *Layer 2 VPNs Over Tunnels*

## RELATED DOCUMENTATION

*Supported Carrier-of-Carriers and Interprovider VPN Standards*

---

*Supported VPWS Standards*

---

*Supported Layer 3 VPN Standards*

---

*Supported Multicast VPN Standards*

---

*Supported VPLS Standards*

---

*Accessing Standards Documents on the Internet*

# Pinging VPNs

## IN THIS CHAPTER

- Pinging VPNs, VPLS, and Layer 2 Circuits | 6
- Pinging a Layer 2 VPN | 7
- Pinging a Layer 2 Circuit | 7

## Pinging VPNs, VPLS, and Layer 2 Circuits

For testing purposes, you can ping Layer 2 VPNs, Layer 3 VPNs, and Layer 2 circuits by using the **ping mpls** command. The **ping mpls** command helps to verify that a VPN or circuit has been enabled and tests the integrity of the VPN or Layer 2 circuit connection between the PE routers. It does not test the connection between a PE router and a CE router. To ping a VPLS routing instance, you issue a **ping vpls instance** command (see *Pinging a VPLS Routing Instance*).

You issue the **ping mpls** command from the ingress PE router of the VPN or Layer 2 circuit to the egress PE router of the same VPN or Layer 2 circuit. When you execute the **ping mpls** command, echo requests are sent as MPLS packets.

The payload is a User Datagram Protocol (UDP) packet forwarded to the address **127.0.0.1**. The contents of this packet are defined in RFC 4379, *Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures*. The label and interface information for building and sending this information as an MPLS packet is the same as for standard VPN traffic, but the time-to-live (TTL) of the innermost label is set to 1.

When the echo request arrives at the egress PE router, the contents of the packet are checked, and then a reply that contains the correct return is sent by means of UDP. The PE router sending the echo request waits to receive an echo reply after a timeout of 2 seconds (you cannot configure this value).

You must configure MPLS at the **[edit protocols mpls]** hierarchy level on the egress PE router (the router receiving the MPLS echo packets) to be able to ping the VPN or Layer 2 circuit. You must also configure the address **127.0.0.1/32** on the egress PE router's **lo0** interface. If this is not configured, the egress PE router does not have this forwarding entry and therefore simply drops the incoming MPLS pings.

The **ping mpls** command has the following limitations:

- You cannot ping an IPv6 destination prefix.

- You cannot ping a VPN or Layer 2 circuit from a router that is attempting a graceful restart.
- You cannot ping a VPN or Layer 2 circuit from a logical system.

You can also determine whether an LSP linking two PE routers in a VPN is up by pinging the end point address of the LSP. The command you use to ping an MPLS LSP end point is **ping mpls lsp-end-point address**. This command tells you what type of LSP (RSVP or LDP) terminates at the address specified and whether that LSP is up or down.

For a detailed description of this command, see the *Junos Routing Protocols and Policies Command Reference*.

## Pinging a Layer 2 VPN

To ping a Layer 2 VPN, use one of the following commands:

- **ping mpls l2vpn interface *interface-name***

You ping an interface configured for the Layer 2 VPN on the egress PE router.

- **ping mpls l2vpn instance *l2vpn-instance-name* local-site-id *local-site-id-number* remote-site-id *remote-site-id-number***

You ping a combination of the Layer 2 VPN routing instance name, the local site identifier, and the remote site identifier to test the integrity of the Layer 2 VPN connection (specified by the identifiers) between the ingress and egress PE routers.

### RELATED DOCUMENTATION

[Example: Configuring MPLS-Based Layer 2 VPNs | 16](#)

## Pinging a Layer 2 Circuit

To ping a Layer 2 circuit, use one of the following commands:

- **ping mpls l2circuit interface *interface-name***

You ping an interface configured for the Layer 2 circuit on the egress PE router.

- **ping mpls l2circuit virtual-circuit neighbor *<prefix>* *<virtual-circuit-id>***

You ping a combination of the IPv4 prefix and the virtual circuit identifier on the egress PE router to test the integrity of the Layer 2 circuit between the ingress and egress PE routers.

# Layer 2 VPNs Configuration Overview

## IN THIS CHAPTER

- Introduction to Configuring Layer 2 VPNs | 8
- Configuring the Local Site on PE Routers in Layer 2 VPNs | 10
- Example: Configuring MPLS-Based Layer 2 VPNs | 16

## Introduction to Configuring Layer 2 VPNs

To configure Layer 2 virtual private network (VPN) functionality, you must enable Layer 2 VPN support on the provider edge (PE) router. You must also configure PE routers to distribute routing information to the other PE routers in the VPN and configure the circuits between the PE routers and the customer edge (CE) routers.

Each Layer 2 VPN is configured under a routing instance of type **l2vpn**. An **l2vpn** routing instance can transparently carry Layer 3 traffic across the service provider's network. As with other routing instances, all logical interfaces belonging to a Layer 2 VPN routing instance are listed under that instance.

The configuration of the CE routers is not relevant to the service provider. The CE routers need to provide only appropriate Layer 2 circuits (with appropriate circuit identifiers, such as data-link connection identifier [DLCI], virtual path identifier/virtual channel identifier [VPI/VCI], or virtual LAN [VLAN] ID) to send traffic to the PE router.

To configure Layer 2 VPNs, include the following statements:

**NOTE:** On the EX9200 switches, replace **encapsulation-type** with the **encapsulation** statement.

```
description text;  
instance-type l2vpn;  
interface interface-name;  
route-distinguisher (as-number:id| ip-address:id);  
vrf-export [ policy-names ];
```



```

vrf-import [ policy-names ];
vrf-target {
    community;
    import community-name;
    export community-name;
}
protocols {
    l2vpn {
        (control-word | no-control-word);
        encapsulation-type type;
        site site-name {
            interface interface-name {
                description text;
                remote-site-id remote-site-id;
            }
            site-identifier identifier;
            site-preference preference-value {
                backup;
                primary;
            }
        }
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
}

```

You can include these statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

For Layer 2 VPNs, only some of the statements in the [edit routing-instances] hierarchy are valid. For the full hierarchy, see *Junos OS Routing Protocols Library*.

In addition to these statements, you must configure MPLS label-switched paths (LSPs) between the PE routers, IBGP sessions between the PE routers, and an interior gateway protocol (IGP) on the PE and provider (P) routers. You must also configure the statements that are required for all types of VPN configuration.

By default, Layer 2 VPNs are disabled.

Many of the configuration procedures for Layer 2 VPNs are identical to the procedures for Layer 3 VPNs and virtual private LAN service (VPLS).

## Configuring the Local Site on PE Routers in Layer 2 VPNs

### IN THIS SECTION

- [Configuring a Layer 2 VPN Routing Instance | 10](#)
- [Configuring the Site | 11](#)
- [Configuring the Remote Site ID | 12](#)
- [Configuring the Encapsulation Type | 13](#)
- [Configuring a Site Preference and Layer 2 VPN Multihoming | 14](#)
- [Tracing Layer 2 VPN Traffic and Operations | 15](#)

For each local site, the PE router advertises a set of VPN labels to the other PE routers servicing the Layer 2 VPN. The VPN labels constitute a single block of contiguous labels; however, to allow for reprovisioning, more than one such block can be advertised. Each label block consists of a label base, a range (the size of the block), and a remote site ID that identifies the sequence of remote sites that connect to the local site using this label block (the remote site ID is the first site identifier in the sequence). The encapsulation type is also advertised along with the label block.

The following sections explain how to configure the connections to the local site on the PE router.

**NOTE:** Not all subtasks are supported on all platforms; check the CLI on your device.

### Configuring a Layer 2 VPN Routing Instance

To configure a Layer 2 VPN on your network, configure a Layer 2 VPN routing instance on the PE router by including the **l2vpn** statement:

**NOTE:** On the EX9200 switches, replace **encapsulation-type** with the **encapsulation** statement.

```
l2vpn {  
  (control-word | no-control-word);  
  encapsulation-type type;
```

```

traceoptions {
  file filename <files number> <size size> <world-readable | no-world-readable>;
  flag flag <flag-modifier> <disable>;
}
site site-name {
  site-identifier identifier;
  site-preference preference-value {
    backup;
    primary;
  }
  interface interface-name {
    description text;
    remote-site-id remote-site-id;
  }
}
}

```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

**NOTE:** You cannot configure a routing protocol (OSPF, RIP, IS-IS or BGP) inside a Layer 2 VPN routing instance (**instance-type l2vpn**). The Junos CLI disallows this configuration.

Instructions for how to configure the remaining statements are included in the sections that follow.

## Configuring the Site

All the Layer 2 circuits provisioned for a local site are listed as the set of logical interfaces (specified by including the **interface** statement) within the **site** statement.

On each PE router, you must configure each site that has a circuit to the PE router. To do this, include the **site** statement:

```

site site-name {
  site-identifier identifier;
  site-preference preference-value {
    backup;
    primary;
  }
}

```

```

interface interface-name {
    description text;
    remote-site-id remote-site-ID;
}

```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols l2vpn]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols l2vpn]

You must configure the following for each site:

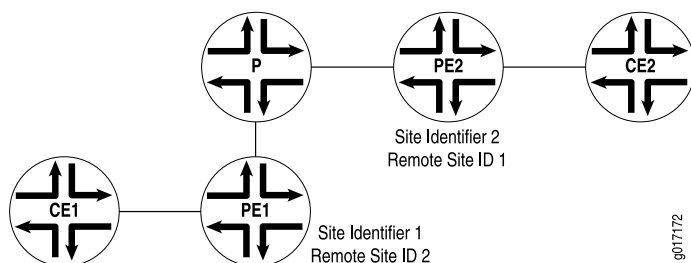
- **site-name**—Name of the site.
- **site-identifier identifier**—Unsigned 16-bit number greater than zero that uniquely identifies the local Layer 2 VPN site. The site identifier corresponds to the remote site ID configured on another site within the same VPN.
- **interface interface-name**—The name of the interface and, optionally, a remote site ID for remote site connections. See [“Configuring the Remote Site ID” on page 12](#).

## Configuring the Remote Site ID

The remote site ID allows you to configure a sparse Layer 2 VPN topology. A sparse topology means that each site does not have to connect to all the other sites in the VPN; thus it is unnecessary to allocate circuits for all the remote sites. Remote site IDs are particularly important if you configure a topology more complicated than full-mesh, such as a hub-and-spoke topology.

The remote site ID (configured with the **remote-site-id** statement) corresponds to the site ID (configured with the **site-identifier** statement) configured at a separate site. [Figure 2 on page 12](#) illustrates the relationship between the site identifier and the remote site ID.

Figure 2: Relationship Between the Site Identifier and the Remote Site ID



As illustrated by the figure, the configuration for Router PE1 connected to Router CE1 is as follows:

```

site-identifier 1;
interface so-0/0/0 {
    remote-site-id 2;
}

```

The configuration for Router PE2 connected to Router CE2 is as follows:

```

site-identifier 2;
interface so-0/0/1 {
    remote-site-id 1;
}

```

The remote site ID (2) on Router PE1 corresponds to the site identifier (2) on Router PE2. On Router PE2, the remote site ID (1) corresponds to the site identifier (1) on Router PE1.

To configure the remote site ID, include the **remote-site-id** statement:

```
remote-site-id remote-site-id;
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols l2vpn site *site-name* interface *interface-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols l2vpn site *site-name* interface *interface-name*]

If you do not explicitly include the **remote-site-id** statement for the interface configured at the [edit routing-instances *routing-instance-name* protocols l2vpn site *site-name*] hierarchy level, a remote site ID is assigned to that interface.

The remote site ID for an interface is automatically set to 1 higher than the remote site ID for the previous interface. The order of the interfaces is based on their **site-identifier** statements. For example, if the first interface in the list does not have a remote site ID, its ID is set to 1. The second interface in the list has its remote site ID set to 2, and the third has its remote site ID set to 3. The remote site IDs of any interfaces that follow are incremented in the same manner if you do not explicitly configure them.

## Configuring the Encapsulation Type

The encapsulation type you configure at each Layer 2 VPN site varies depending on which Layer 2 protocol you choose to configure. If you configure **ethernet-vlan** as the encapsulation type, you need to use the same protocol at each Layer 2 VPN site.

You do not need to use the same protocol at each Layer 2 VPN site if you configure any of the following encapsulation types:

- **atm-aal5**—Asynchronous Transfer Mode (ATM) Adaptation Layer (AAL5)
- **atm-cell**—ATM cell relay
- **atm-cell-port-mode**—ATM cell relay port promiscuous mode
- **atm-cell-vc-mode**—ATM virtual circuit (VC) cell relay nonpromiscuous mode
- **atm-cell-vp-mode**—ATM virtual path (VP) cell relay promiscuous mode
- **cisco-hdlc**—Cisco Systems-compatible High-Level Data Link Control (HDLC)
- **ethernet**—Ethernet
- **ethernet-vlan**—Ethernet virtual LAN (VLAN)
- **frame-relay**—Frame Relay
- **frame-relay-port-mode**—Frame Relay port mode
- **interworking**—Layer 2.5 interworking VPN
- **ppp**—Point-to-Point Protocol (PPP)

If you configure different protocols at your Layer 2 VPN sites, you need to configure a translational cross-connect (TCC) encapsulation type. For more information, see [“Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits” on page 35](#).

To configure the Layer 2 protocol accepted by the PE router, specify the encapsulation type by including the **encapsulation-type** statement:

```
encapsulation-type type;
```

For EX9200 switches, specify the encapsulation type by including the **encapsulation** statement:

```
encapsulation type;
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols l2vpn]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols l2vpn]

## Configuring a Site Preference and Layer 2 VPN Multihoming

You can specify the preference value advertised for a particular Layer 2 VPN site. The site preference value is encoded in the BGP local preference attribute. When a PE router receives multiple advertisements with the same CE device identifier, the advertisement with the highest local preference value is preferred.

You can also use the **site-preference** statement to enable multihoming for Layer 2 VPNs. Multihoming allows you to connect a CE device to multiple PE routers. In the event that a connection to the primary PE router fails, traffic can be automatically switched to the backup PE router.

To configure a site preference for a Layer 2 VPN, include the **site-preference** statement:

```
site-preference preference-value {
    backup;
    primary;
}
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols l2vpn site *site-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols l2vpn site *site-name*]

You can also specify either the **backup** option or the **primary** option for the **site-preference** statement. The backup option specifies the preference value as 1, the lowest possible value, ensuring that the Layer 2 VPN site is the least likely to be selected. The primary option specifies the preference value as 65,535, the highest possible value, ensuring that the Layer 2 VPN site is the most likely to be selected.

For Layer 2 VPN multihoming configurations, specifying the **primary** option for a Layer 2 VPN site designates the connection from the PE router to the CE device as the preferred connection if the CE device is also connected to another PE router. Specifying the **backup** option for a Layer 2 VPN site designates the connection from the PE router to the CE device as the secondary connection if the CE device is also connected to another PE router.

## Tracing Layer 2 VPN Traffic and Operations

To trace Layer 2 VPN protocol traffic, specify options for the **traceoptions** statement in the Layer 2 VPN configuration:

```
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols l2vpn]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols l2vpn]

The following trace flags display the operations associated with Layer 2 VPNs:

- **all**—All Layer 2 VPN tracing options.
- **connections**—Layer 2 connections (events and state changes).
- **error**—Error conditions.
- **general**—General events.
- **nlri**—Layer 2 advertisements received or sent by means of the BGP.
- **normal**—Normal events.
- **policy**—Policy processing.
- **route**—Routing information.
- **state**—State transitions.
- **task**—Routing protocol task processing.
- **timer**—Routing protocol timer processing.
- **topology**—Layer 2 VPN topology changes caused by reconfiguration or advertisements received from other PE routers using BGP.

#### *Disabling Normal TTL Decrementing for VPNs*

To diagnose networking problems related to VPNs, it can be useful to disable normal time-to-live (TTL) decrementing. In Junos, you can do this with the **no-propagate-ttl** and **no-decrement-ttl** statements. However, when you are tracing VPN traffic, only the **no-propagate-ttl** statement is effective.

For the **no-propagate-ttl** statement to have an effect on VPN behavior, you need to clear the PE-router-to-PE-router BGP session, or disable and then enable the VPN routing instance.

For more information about the **no-propagate-ttl** and **no-decrement-ttl** statements, see the *MPLS Applications User Guide*.

## Example: Configuring MPLS-Based Layer 2 VPNs

### IN THIS SECTION

- [Requirements | 17](#)
- [Overview and Topology | 17](#)
- [Configuring the Local PE Routing Device | 21](#)
- [Configuring the Remote PE Routing Device | 24](#)
- [Verification | 28](#)



You can implement an MPLS-based Layer 2 virtual private network (VPN) using Junos OS routing devices to interconnect customer sites with Layer 2 technology. Layer 2 VPNs give customers complete control of their own routing. To support an MPLS-based Layer 2 VPN, you need to add components to the configuration of the two provider edge (PE) routing devices. You do not need to change the configuration of the provider devices.

This example shows how to configure an MPLS-based Layer 2 VPN.

**NOTE:** You can configure both an MPLS-based Layer 2 VPN and an MPLS-based Layer 3 VPN on the same device. However, you cannot configure the same customer edge interface to support both a Layer 2 VPN and a Layer 3 VPN. The core interfaces and the loopback interfaces are configured in the same way for Layer 2 VPNs and Layer 3 VPNs.

## Requirements

This example uses the following hardware and software components:

- Junos OS Release 11.1 or later if you are using EX Series switches
- Two PE routing devices

Before you configure the Layer 2 VPN components, configure the basic components for an MPLS network:

- Configure two PE routing devices. See *Configuring MPLS on Provider Edge EX8200 and EX4500 Switches Using Circuit Cross-Connect*.
- Configure one or more provider devices. See *Configuring MPLS on EX8200 and EX4500 Provider Switches*.

**NOTE:** A Layer 2 VPN requires that the PE routing devices be configured using circuit cross-connect (CCC). The provider routing devices are configured in the same way for MPLS using CCC and for IP over MPLS.

## Overview and Topology

A Layer 2 VPN provides complete separation between the provider's network and the customer's network—that is, the PE devices and the CE devices do not exchange routing information. Some benefits of a Layer 2 VPN are that it is private, secure, and flexible.

This example shows how to configure Layer 2 VPN components on the local and remote PE devices. This example does not include configuring a provider device, because there are no specific Layer 2 VPN components on the provider devices.

In the basic MPLS configuration of the PE devices using a circuit cross-connect (CCC), the PE devices are configured to use an interior gateway protocol (IGP), such as OSPF or IS-IS, as the routing protocol between the MPLS devices and LDP or RSVP as the signaling protocol. Traffic engineering is enabled. A label-switched path (LSP) is configured within the **[edit protocols]** hierarchy. However, unlike the basic MPLS configuration using a CCC, you do not need to associate the LSP with the customer edge interface. When you are configuring a Layer 2 VPN, you must use BGP signaling. The BGP signaling automates the connections, so manual configuration of the association between the LSP and the customer edge interface is not required.

The following components must be added to the PE routing devices for an MPLS-based Layer 2 VPN:

- BGP group with **family l2vpn signaling**
- Routing instance using instance type **l2vpn**
- The physical layer encapsulation type (**ethernet**) must be specified on the customer edge interface and the encapsulation type must also be specified in the configuration of the routing instance.

Figure 3 on page 18 illustrates the topology of this MPLS-based Layer 2 VPN.

Figure 3: MPLS-Based Layer 2 VPN

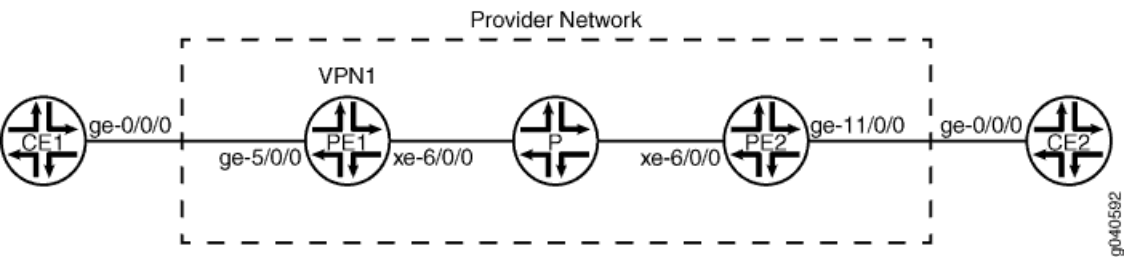


Table 3 on page 18 shows the settings of the customer edge interface on the local CE device.

Table 3: Local CE Routing Device in the MPLS-Based Layer 2 VPN Topology

Property	Settings	Description
Local CE routing device hardware	Routing device	CE1
Customer edge interface	<b>ge-0/0/0 unit 0</b> <b>family inet</b> <b>address 10.0.0.2/16</b>	Interface that connects CE1 to PE1.

Table 4 on page 18 shows the settings of the customer edge interface on the remote CE routing device.

Table 4: Remote CE Routing Device in the MPLS-Based Layer 2 VPN Topology

Property	Settings	Description
Remote CE routing device hardware	Routing device	CE2

Table 4: Remote CE Routing Device in the MPLS-Based Layer 2 VPN Topology (*continued*)

Property	Settings	Description
Customer edge interface	<b>ge-0/0/0 unit 0</b> <b>family inet</b> <b>address 10.0.0.1/16</b>	Interface that connects CE2 to PE2.

Table 5 on page 19 shows the Layer 2 VPN components of the local PE routing device.

Table 5: Layer 2 VPN Components of the Local PE Routing Device

Property	Settings	Description
Local PE routing device hardware	Routing device	PE1
Customer edge interface	<b>ge-5/0/0</b> <b>encapsulation ethernet-ccc</b> <b>unit 0</b> <b>family ccc</b>	Connects PE1 to CE1.  For the Layer 2 VPN, add <b>ethernet-ccc</b> as the physical layer encapsulation type.  <b>NOTE:</b> The <b>family ccc</b> should already have been completed as part of the basic MPLS configuration of a PE routing device for circuit cross-connect. It is included here to show what was specified for that portion of the configuration.
Core interface	<b>xe-6/0/0 unit 0</b> <b>family inet address 10.0.0.60/16</b> <b>family iso</b> <b>family mpls</b>	Connects PE1 to P.  <b>NOTE:</b> This portion of the configuration should already have been completed as part of the basic MPLS configuration. It is included here to show what was specified for that portion of the configuration.
Loopback interface	<b>lo0 unit 0</b> <b>family inet address 192.0.2.0/24</b> <b>family iso address 49.0001.2102.2021.0210.00</b>	<b>NOTE:</b> This portion of the configuration should already have been completed as part of the basic MPLS configuration. It is included here to show what was specified for that portion of the configuration.

Table 5: Layer 2 VPN Components of the Local PE Routing Device (*continued*)

Property	Settings	Description
BGP	<b>bgp</b>	Added for the Layer 2 VPN configuration.
Routing instance	<b>vpn1</b>	Added for the Layer 2 VPN configuration

[Table 6 on page 20](#) shows the Layer 2 VPN components of the remote PE routing device.

Table 6: Layer 2 VPN Components of the Remote PE Routing Device

Property	Settings	Description
PE routing device hardware	Routing device	PE2
Customer edge interface	<b>ge-11/0/0</b> <b>encapsulation ethernet-ccc unit 0</b> <b>family ccc</b>	Connects PE2 to CE2.  For the Layer 2 VPN, add <b>ethernet-ccc</b> as the physical layer encapsulation type.  <b>NOTE:</b> The <b>family ccc</b> should already have been completed as part of the basic MPLS configuration of a PE routing device for circuit cross-connect. It is included here to show what was specified for that portion of the configuration.
Core interface	<b>xe-6/0/0</b>  <b>unit 0</b> <b>family inet</b> <b>address 10.2.0.61/16 family iso</b> <b>family mpls</b>	Connects PE2 to P.  <b>NOTE:</b> This portion of the configuration should already have been completed as part of the basic MPLS configuration. It is included here to show what was specified for that portion of the configuration.
Loopback interface	<b>lo0 unit 0</b> <b>family inet address 192.0.2.3/24</b> <b>family iso address</b> <b>49.0001.2202.2022.0220.00</b>	<b>NOTE:</b> This portion of the configuration should already have been completed as part of the basic MPLS configuration. It is included here to show what was specified for that portion of the configuration.

Table 6: Layer 2 VPN Components of the Remote PE Routing Device (*continued*)

Property	Settings	Description
BGP	<b>bgp</b>	Added for the Layer 2 VPN configuration.
Routing instance	<b>vpn1</b>	Added for the Layer 2 VPN configuration.

## Configuring the Local PE Routing Device

### CLI Quick Configuration

To quickly configure the Layer 2 VPN components on the local PE routing device, copy the following commands and paste them into the routing device terminal window:

```
[edit]
set interfaces ge-5/0/0 encapsulation ethernet-ccc
set protocols bgp group ibgp local-address 192.0.2.0 family l2vpn signaling
set protocols bgp group ibgp type internal
set protocols bgp group ibgp neighbor 192.0.2.3
set routing-instances vpn1 instance-type l2vpn
set routing-instances vpn1 interface ge-5/0/0
set routing-instances vpn1 route-distinguisher 192.0.2.0:21
set routing-instances vpn1 vrf-target target:21:21
set routing-instances vpn1 protocols l2vpn encapsulation-type ethernet
set routing-instances vpn1 protocols l2vpn interface ge-5/0/0.0 description "BETWEEN PE1 AND CE1"
set routing-instances vpn1 protocols l2vpn site JE-V21 site-identifier 21 interface ge-5/0/0 remote-site-id
26
```

### Step-by-Step Procedure

To configure the Layer 2 VPN components on the local PE routing device:

1. Configure the customer edge interface to use the physical encapsulation type **ethernet-ccc**:

```
[edit]
user@PE1# set interfaces ge-5/0/0 encapsulation ethernet-ccc
```

2. Configure BGP, specifying the loopback address as the local address and enabling **family l2vpn signaling**:

```
[edit protocols bgp]
user@PE1# set group ibgp local-address 192.0.2.0 family l2vpn signaling
```

3. Configure the BGP group, specifying the group name and type:

```
[edit protocols bgp]
user@PE1# set group ibgp type internal
```

4. Configure the BGP neighbor, specifying the loopback address of the remote PE routing device as the neighbor's address:

```
[edit protocols bgp]
user@PE1# set group ibgp neighbor 192.0.2.3/24
```

5. Configure the routing instance, specifying the routing-instance name and using **l2vpn** as the instance type:

```
[edit routing-instances]
user@PE1# set vpn1 instance-type l2vpn
```

6. Configure the routing instance to apply to the customer edge interface:

```
[edit routing-instances]
user@PE1# set vpn1 interface ge-5/0/0
```

7. Configure the routing instance to use a route distinguisher:

```
[edit routing-instances]
user@PE1# set vpn1 route-distinguisher 192.0.2.0:21
```

8. Configure the VPN routing and forwarding (VRF) target of the routing instance:

```
[edit routing-instances]
user@PE1# set vpn1 vrf-target target:21:21
```

**NOTE:** You can create more complex policies by explicitly configuring VRF import and export policies using the import and export options. See the [Junos OS VPNs Configuration Guide](#).

9. Configure the protocols and encapsulation type used by the routing instance:

```
[edit routing-instances]
user@PE1# set vpn1 protocols l2vpn encapsulation-type ethernet
```

10. Apply the routing instance to a customer edge interface and specify a description for it:

```
[edit routing-instances]
```

```
user@PE1# set vpn1 protocols interface ge-5/0/0.0 description "BETWEEN PE1 AND CE1"
```

11. Configure the routing-instance protocols site:

```
[edit routing-instances]
```

```
user@PE1# set vpn1 protocols l2vpn site JE-V21 site-identifier 21 remote-site-id 26
```

**NOTE:** The remote site ID (configured with the **remote-site-id** statement) corresponds to the site ID (configured with the **site-identifier** statement) configured on the other PE routing device.

## Results

Display the results of the configuration:

```
user@PE1# show
```

```
interfaces {
  ge-5/0/0 {
    encapsulation ethernet-ccc;
    unit 0 {
      family ccc;
    }
  }
  xe-6/0/0 {
    unit 0 {
      family inet {
        address 10.0.0.60/16;
      }
      family mpls;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 192.0.2.0/24;
      }
      family iso {
        address 49.0001.2102.2021.0210.00;
      }
    }
  }
}
```

```

}
protocols {
  rsvp {
    interface lo0.0;
    interface xe-0/0/6.0;
  }
  mpls {
    label-switched-path lsp_to_pe2 {
      to 192.0.2.3;
    }
  }
  bgp {
    group ibgp
    type internal
    local-address 192.0.2.0
    family l2vpn signaling
  }
}
routing-instances {
  vpn1 {
    instance-type l2vpn;
    interface ge-5/0/0.0;
    route-distinguisher 192.0.2.0:21;
    vrf-target target:21:21;
    protocols {
      l2vpn {
        encapsulation-type ethernet;
        interface ge-5/0/0.0 {
          description "BETWEEN PE1 AND CE1";
        }
        site JE-V21 {
          site-identifier 21;
          interface ge-5/0/0.0 {
            remote-site-id 26;
          }
        }
      }
    }
  }
}
}
}
}

```

## Configuring the Remote PE Routing Device

### CLI Quick Configuration



To quickly configure the Layer 2 VPN components on the remote PE routing device, copy the following commands and paste them into the routing device terminal window:

```
[edit]
set interfaces ge-11/0/0 encapsulation ethernet-ccc
set protocols bgp group ibgp local-address 192.0.2.3 family l2vpn signaling
set protocols bgp group ibgp type internal
set protocols bgp group ibgp neighbor 192.0.2.0
set routing-instances vpn1 instance-type l2vpn
set routing-instances vpn1 interface ge-11/0/0
set routing-instances vpn1 route-distinguisher 192.0.2.0:21
set routing-instances vpn1 vrf-target target:21:21
set routing-instances vpn1 protocols l2vpn encapsulation-type ethernet
set routing-instances vpn1 protocols l2vpn interface ge-11/0/0.0 description "BETWEEN PE1 AND CE1"
set routing-instances vpn1 protocols l2vpn site T26-VPN1 site-identifier 26 remote-site-id 21
```

### Step-by-Step Procedure

To configure the Layer 2 VPN components on the remote PE routing device:

1. Configure the customer edge interface to use the physical encapsulation type **ethernet-ccc**:

```
[edit]
user@PE1# set interfaces ge-11/0/0 encapsulation ethernet-ccc
```

2. Configure BGP, specifying the loopback address as the **local-address** and specifying **family l2vpn signaling**:

```
[edit protocols bgp]
user@PE2# set group ibgp local-address 192.0.2.3 family l2vpn signaling
```

3. Configure the BGP group, specifying the group name and type:

```
[edit protocols bgp]
user@PE2# set group ibgp type internal
```

4. Configure the BGP neighbor, specifying the loopback address of the remote PE routing device as the neighbor's address:

```
[edit protocols bgp]
user@PE2# set group ibgp neighbor 192.0.2.0
```

5. Configure the routing instance, specifying the routing-instance name and using **l2vpn** as the **instance-type**:

```
[edit routing-instances]
user@PE2# set vpn1 instance-type l2vpn
```

6. Configure the routing instance to apply to the customer edge interface:

```
[edit routing-instances]
user@PE2# set vpn1 interface ge-11/0/0.0
```

7. Configure the routing instance to use a route distinguisher, using the format *ip-address:number*:

```
[edit routing-instances]
user@PE2# set vpn1 route-distinguisher 192.0.2.0:21
```

8. Configure the VPN routing and forwarding (VRF) target of the routing instance:

```
[edit routing-instances]
user@PE2# set vpn1 vrf-target target:21:21
```

9. Configure the protocols and encapsulation type used by the routing instance:

```
[edit routing-instances]
user@PE2# set vpn1 protocols l2vpn encapsulation-type ethernet
```

10. Apply the routing instance to a customer edge interface and specify a description for it:

```
[edit routing-instances]
user@PE1# set vpn1 protocols interface ge-11/0/0.0 description "BETWEEN PE1 AND CE1"
```

11. Configure the routing-instance protocols site:

```
[edit routing-instances]
user@PE2# set vpn1 protocols l2vpn site T26-VPN1 site-identifier 26 remote-site-id 21
```

**NOTE:** The remote site ID (configured with the **remote-site-id** statement) corresponds to the site ID (configured with the **site-identifier** statement) configured on the other PE routing device.

## Results

Display the results of the configuration:

```
user@PE2# show
```

```

interfaces {
  ge-11/0/0 {
    encapsulation ethernet-ccc;
    unit 0 {
      family ccc;
    }
  }
  xe-6/0/0 {
    unit 0 {
      family inet {
        address 10.2.0.61/16;
      }
      family mpls;
    }
  }
  lo0 {
    unit 0 {
      family inet {
        address 192.0.2.3/24;
      }
      family iso {
        address 49.0001.2202.2022.0220.00;
      }
    }
  }
}

protocols {
  rsvp {
    interface lo0.0;
    interface xe-0/0/6.0;
  }
  mpls {
    label-switched-path lsp_to_pe1 {
      to 192.0.2.0;
    }
  }
  bgp {
    group ibgp
    type internal
    local-address 192.0.2.0
    family l2vpn signaling
  }
}

routing-instances {

```

```

vpn1 {
  instance-type l2vpn;
  interface ge-11/0/0.0;
  route-distinguisher 192.0.2.0:21;
  vrf-target target:21:21;
  protocols {
    l2vpn {
      encapsulation-type ethernet;
      interface ge-11/0/0.0 {
        description "BETWEEN PE1 AND CE1";
      }
      site T26-VPN1 {
        site-identifier 26;
        interface ge-11/0/0.0 {
          remote-site-id 21;
        }
      }
    }
  }
}

```

## Verification

### IN THIS SECTION

- [Verifying the Layer 2 VPN Connection | 28](#)
- [Verifying the Status of MPLS Label-Switched Paths | 29](#)
- [Verifying BGP Status | 30](#)
- [Verifying the Status of the RSVP Sessions | 31](#)
- [Verifying the Routes in the Routing Table | 31](#)
- [Pinging the Layer 2 VPN Connections | 32](#)

To confirm that the MPLS-based Layer 2 VPN is working properly, perform these tasks:

### ***Verifying the Layer 2 VPN Connection***

#### **Purpose**

Verify that the Layer 2 VPN connection is up.

## Action

user@PE1> **show l2vpn connections**

Layer-2 VPN connections:

Legend for connection status (St)

EI -- encapsulation invalid	NC -- interface encapsulation not CCC/TCC/VPLS
EM -- encapsulation mismatch	WE -- interface and instance encaps not same
VC-Dn -- Virtual circuit down	NP -- interface hardware not present
CM -- control-word mismatch	-> -- only outbound connection is up
CN -- circuit not provisioned	<- -- only inbound connection is up
OR -- out of range	Up -- operational
OL -- no outgoing label	Dn -- down
LD -- local site signaled down	CF -- call admission control failure
RD -- remote site signaled down	SC -- local and remote site ID collision
LN -- local site not designated	LM -- local site ID not minimum designated
RN -- remote site not designated	RM -- remote site ID not minimum designated
XX -- unknown connection status	IL -- no incoming label
MM -- MTU mismatch	MI -- Mesh-Group ID not available
BK -- Backup connection	ST -- Standby connection
PF -- Profile parse failure	PB -- Profile busy
RS -- remote site standby	SN -- Static Neighbor

Legend for interface status

Up -- operational  
Dn -- down

Instance: vpn1

Local site: JE-V21 (21)

connection-site	Type	St	Time last up	# Up trans
26	rmt	Up	Apr 16 05:53:21 2010	1
Remote PE: 192.0.2.3, Negotiated control-word: Yes (Null)				
Incoming label: 800000, Outgoing label: 800001				
Local interface: ge-5/0/0.0, Status: Up, Encapsulation: ETHERNET				

## Meaning

The **St** field in the output shows that the Layer 2 VPN connection to **Remote PE (192.0.2.3)** is up.

## Verifying the Status of MPLS Label-Switched Paths

## Purpose

Verify that the MPLS label-switched paths (ingress and egress) are up.

### Action

```
user@PE1> show mpls lsp
```

```
Ingress LSP: 1 sessions
```

To	From	State	Rt	P	ActivePath	LSPname
192.0.2.3	192.0.2.0	Up	0	*		lsp_to_pe2

```
Total 1 displayed, Up 1, Down 0
```

```
Egress LSP: 1 sessions
```

To	From	State	Rt	Style	Labelin	Labelout	LSPname
192.0.2.0	192.0.2.3	Up	0	1 FF	3	-	lsp_to_pe1

```
Total 1 displayed, Up 1, Down 0
```

```
Transit LSP: 0 sessions
```

```
Total 0 displayed, Up 0, Down 0
```

### Meaning

The **State** field in the output shows that the Ingress LSP to **Remote PE (192.0.2.3)** is up, and the Egress LSP from the remote PE routing device to this PE routing device (**192.0.2.0**) is also up.

### Verifying BGP Status

#### Purpose

Verify that BGP is up.

### Action

```
user@PE1> show bgp summary
```

```
Groups: 1 Peers: 1 Down peers: 0
```

Table	Tot Paths	Act Paths	Suppressed	History	Damp	State	Pending
bgp.l2vpn.0	1	1	0	0		0	0

Peer	AS	InPkt	OutPkt	OutQ	Flaps	Last Up/Dwn
------	----	-------	--------	------	-------	-------------

```
State|#Active/Received/Accepted/Damped...
```

```
192.0.2.3          10          33          34          0          1          13:24 Establ
```

```
bgp.l2vpn.0: 1/1/1/0
```

```
vpn2.l2vpn.0: 1/1/1/0
```

### Meaning

The output shows that the remote PE routing device (**192.0.2.3**) is listed as the BGP peer and that a protocol session has been established. It also shows the number of packets received from the remote PE routing device (**33**) and the number of packets sent (**34**) to the remote PE routing device.

### Verifying the Status of the RSVP Sessions

#### Purpose

Verify that the RSVP sessions (ingress and egress) are up.

#### Action

```
user@PE1> show rsvp session
```

```
Ingress RSVP: 1 sessions
```

To	From	State	Rt	Style	Labelin	Labelout	LSPname
192.0.2.3	192.0.2.0	Up	0	1 FF	-	462880	lsp_to_pe2

Total 1 displayed, Up 1, Down 0

```
Egress RSVP: 1 sessions
```

To	From	State	Rt	Style	Labelin	Labelout	LSPname
192.0.2.0	192.0.2.3	Up	0	1 FF	3	-	lsp_to_pe1

Total 1 displayed, Up 1, Down 0

```
Transit RSVP: 0 sessions
```

Total 0 displayed, Up 0, Down 0

### Meaning

The output shows that both the ingress RSVP session and the egress RSVP session are up.

### Verifying the Routes in the Routing Table

#### Purpose

On routing device PE 1, use the **show route table** command to verify that the routing table is populated with the Layer 2 VPN routes used to forward the traffic.

## Action

user@PE1> **show route table bgp.l2vpn.0**

```

bgp.l2vpn.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192:2:27:27/96
    *[BGP/170] 00:13:55, localpref 100, from 192.0.2.3
        AS path: I
    > to 10.2.0.24 via ge-6/0/46.0, label-switched-path lsp_to_pe2

```

user@PE1> **show route table vpn1.l2vpn.0**

```

vpn1.l2vpn.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192:2:27:27/96
    *[BGP/170] 00:14:00, localpref 100, from 192.0.2.3
        AS path: I
    > to 10.2.0.24 via ge-6/0/46.0, label-switched-path lsp_to_pe2

192:2:28:27/96
    *[L2VPN/170/-101] 00:15:55, metric2 1
        Indirect

```

## Meaning

The command **show route table bgp.l2vpn.0** displays all Layer 2 VPN routes that have been created on this routing device. The command **show route table vpn1.l2vpn.0** shows the Layer 2 VPN routes that have been created for the routing instance **vpn1**.

## Pinging the Layer 2 VPN Connections

### Purpose

Verify connectivity.



## Action

```
user@PE1> ping mpls l2vpn interface xe-6/0/0.0 reply-mode ip-udp
```

```
!!!!
--- lsping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

```
user@PE1> ping mpls l2vpn instance vpn1 remote-site-id 26 local-site-id 21 detail
```

```
Request for seq 1, to interface 68, labels <800001, 100176>
Reply for seq 1, return code: Egress-ok
Request for seq 2, to interface 68, labels <800001, 100176>
Reply for seq 2, return code: Egress-ok
Request for seq 3, to interface 68, labels <800001, 100176>
Reply for seq 3, return code: Egress-ok
Request for seq 4, to interface 68, labels <800001, 100176>
Reply for seq 4, return code: Egress-ok
Request for seq 5, to interface 68, labels <800001, 100176>
Reply for seq 5, return code: Egress-ok

--- lsping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

## Meaning

The output shows that connectivity is established.

## RELATED DOCUMENTATION

---

*Example: Configuring MPLS on EX8200 and EX4500 Switches*

---

*Example: Configuring MPLS-Based Layer 3 VPNs on EX Series Switches*

---

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## Configuring Layer 2 Interfaces

### IN THIS CHAPTER

- Configuring CCC Encapsulation for Layer 2 VPNs | 34
- Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits | 35
- Configuring the MTU for Layer 2 Interfaces | 37
- Disabling the Control Word for Layer 2 VPNs | 38

### Configuring CCC Encapsulation for Layer 2 VPNs

You need to specify a circuit cross-connect (CCC) encapsulation type for each PE-router-to-CE-router interface running a Layer 2 VPN. This encapsulation type should match the encapsulation type configured under the routing instance. For information about how to configure the encapsulation type under the routing instance, see [“Configuring the Encapsulation Type” on page 13](#).

**NOTE:** A Layer 2 VPN or Layer 2 circuit is not supported if the PE-router-to-P-router interface has VLAN-tagging enabled and uses a nonenhanced Flexible PIC Concentrator (FPC).

For Layer 2 VPNs, you need to configure the CCC encapsulation on the logical interface. You also need to configure an encapsulation on the physical interface. The physical interface encapsulation does not have to be a CCC encapsulation. However, it should match the logical interface encapsulation. For example, if you configure an ATM CCC encapsulation type on the logical interface, you should configure a compatible ATM encapsulation on the physical interface.

**NOTE:** The EX9200 switches only use **ethernet** and **ethernet-vlan** encapsulation types.

To configure the CCC encapsulation type, include the **encapsulation-type** statement:

```
encapsulation-type ccc-encapsulation-type;
```

On the EX9200 switches, replace **encapsulation-type** with the **encapsulation** statement:

```
encapsulation ccc-encapsulation;
```

To configure the CCC encapsulation type on the physical interface, include this statement at the following hierarchy levels:

- **[edit interfaces *interface-name*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name*]**

To configure the CCC encapsulation type on the logical interface, include this statement at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]**

You configure the encapsulation type at the **[edit interfaces]** hierarchy level differently from the **[edit routing-instances]** hierarchy level. For example, you specify the encapsulation as **frame-relay** at the **[edit routing-instances]** hierarchy level and as **frame-relay-ccc** at the **[edit interfaces]** hierarchy level.

You can run both standard Frame Relay and CCC Frame Relay on the same device. If you specify Frame Relay encapsulation (**frame-relay-ccc**) for the interface, you should also configure the encapsulation at the **[edit interfaces *interface name* unit *unit-number*]** hierarchy level as **frame-relay-ccc**. Otherwise, the logical interface unit defaults to standard Frame Relay.

## Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits

Also known as Layer 2.5 VPNs, the translation cross-connect (TCC) encapsulation types allow you to configure different encapsulation types at the ingress and egress of a Layer 2 VPN or the ingress and egress of a Layer 2 circuit. For example, a CE router at the ingress of a Layer 2 VPN path can send traffic in a Frame Relay encapsulation. A CE router at the egress of that path can receive the traffic in an ATM encapsulation.

**NOTE:** The EX9200 switches only use **ethernet** and **ethernet-vlan** encapsulation types.

For information about how to configure encapsulations for Layer 2 circuits, see [“Configuring the Interface Encapsulation Type for Layer 2 Circuits” on page 76](#).

The configuration for TCC encapsulation types is similar to the configuration for CCC encapsulation types. For Layer 2 VPNs, you specify a TCC encapsulation type for each PE-router-to-CE-router interface. The

encapsulation type configured for the interface should match the encapsulation type configured under the routing instance. For information about how to configure the encapsulation type under the routing instance, see [“Configuring the Encapsulation Type” on page 13](#).

**NOTE:** Some platform and FPC combinations can not pass TCC encapsulated ISO traffic. See *Platforms/FPCs That Cannot Forward TCC Encapsulated ISO Traffic* for details.

You need to configure the TCC encapsulation on both the physical and logical interfaces. To configure the TCC encapsulation type, include the **encapsulation-type** statement:

```
encapsulation-type tcc-encapsulation-type;
```

On the EX9200 switches, replace **encapsulation-type** with the **encapsulation** statement:

```
encapsulation tcc-encapsulation;
```

To configure the TCC encapsulation type on the physical interface, include this statement at the following hierarchy levels:

- **[edit interfaces *interface-name*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name*]**

To configure the TCC encapsulation type on the logical interface, include this statement at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]**

You configure the encapsulation type at the **[edit interfaces]** hierarchy level differently than at the **[edit routing-instances]** hierarchy level. For example, you specify the encapsulation as **frame-relay** at the **[edit routing-instances]** hierarchy level and as **frame-relay-tcc** at the **[edit interfaces]** hierarchy level.

For Layer 2.5 VPNs employing an Ethernet interface as the TCC router, you can configure an Ethernet TCC or an extended VLAN TCC.

To configure an Ethernet TCC or an extended VLAN TCC, include the **proxy** and **remote** statements:

```
proxy inet-address;
remote (inet-address | mac-address);
```

You can include these statements at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number* family tcc]
- [edit logical-interfaces *logical-interface-name* interfaces *interface-name* unit *logical-unit-number* family tcc]

The **proxy inet-address** address statement defines the IP address for which the TCC router is acting as proxy.

The **remote (inet-address | mac-address)** statement defines the location of the remote router.

Ethernet TCC is supported on interfaces that carry IP version 4 (IPv4) traffic only. However, Ethernet TCC encapsulation is not supported on 8-port, 12-port, and 48-port Fast Ethernet PICs.

## Configuring the MTU for Layer 2 Interfaces

By default, the MTU used to advertise a Layer 2 pseudowire is determined by taking the interface MTU for the associated physical interface and subtracting the encapsulation overhead for sending IP packets based on the encapsulation. However, encapsulations that support multiple logical interfaces (and multiple Layer 2 pseudowires) rely on the same interface MTU (since they are all associated with the same physical interface). This can prove to be a limitation for VLAN Layer 2 pseudowires using the same Ethernet interface or for Layer 2 pseudowire DLCIs using the same Frame Relay interface.

This can also affect multivendor environments. For example, if you have three PE devices supplied by different vendors and one of the devices only supports an MTU of 1500, even if the other devices support larger MTUs you must configure the MTU as 1500 (the smallest MTU of the three PE devices).

You can explicitly configure which MTU is advertised for a Layer 2 pseudowire, even if the Layer 2 pseudowire is sharing a physical interface with other Layer pseudowires. When you explicitly configure an MTU for a Layer 2 pseudowire, be aware of the following:

- For BGP-based applications such as l2vpn and bgp-vpls, the advertised MTU will be zero unless an MTU value is explicitly set at the [edit routing-instances *routing-instance-name* protocols (*l2vpn* | *vpls*) site *site-name*] hierarchy level.
- An explicitly configured MTU is signaled to the remote PE device. The configured MTU is also compared to the MTU received from the remote PE device. If there is a conflict, the Layer 2 pseudowire is taken down.
- If you configure an MTU for an ATM cell relay interface on an ATM II PIC, the configured MTU is used to compute the cell bundle size advertised for that Layer 2 pseudowire, instead of the default interface MTU.
- A configured MTU is used only in the control plane. It is not enforced in the data plane. You need to ensure that the CE device for a given Layer 2 pseudowire uses the correct MTU for data transmission.

The following procedure describes how to configure the MTU for the Layer 2 interface. This information applies to the following Layer 2 technologies:

- Layer 2 VPNs
- Layer 2 Circuits
- VPLS

1. To configure the MTU for a Layer 2 circuit, include the **mtu** statement:

```
mtu mtu-number;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

2. To allow a Layer 2 pseudowire to be established even though the MTU configured on the local PE router does not match the MTU configured on the remote PE router, include the **ignore-mtu-mismatch** statement:

```
ignore-mtu-mismatch;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

## RELATED DOCUMENTATION

[ignore-mtu-mismatch](#) | 205

[mtu](#) | 223

## Disabling the Control Word for Layer 2 VPNs

A 4-byte control word provides support for the emulated VC encapsulation for Layer 2 VPNs. This control word is added between the Layer 2 protocol data unit (PDU) being transported and the VC label that is used for demultiplexing. Various networking formats (ATM, Frame Relay, Ethernet, and so on) use the control word in a variety of ways.

On networks with equipment that does not support the control word, you can disable it by including the **no-control-word** statement:

```
no-control-word;
```

You can include this statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols l2vpn]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols l2vpn]

For more information about configuring the control word, see [“Configuring the Control Word for Layer 2 Circuits” on page 70](#) and the *Layer 2 Circuits User Guide*.

**NOTE:** Use the **no-control-word** statement to disable the control word when the topology uses generic routing encapsulation (GRE) as the connection mechanism between PEs, and one of the PEs is an M Series router.

## RELATED DOCUMENTATION

---

[Configuring the Control Word for Layer 2 Circuits | 70](#)

---

[control-word | 163](#)

---

[l2vpn | 213](#)

# Configuring Path Selection for Layer 2 VPNs and VPLS

## IN THIS CHAPTER

- [Understanding BGP Path Selection | 40](#)
- [Enabling BGP Path Selection for Layer 2 VPNs and VPLS | 46](#)

## Understanding BGP Path Selection

For each prefix in the routing table, the routing protocol process selects a single best path. After the best path is selected, the route is installed in the routing table. The best path becomes the active route if the same prefix is not learned by a protocol with a lower (more preferred) global preference value, also known as the administrative distance. The algorithm for determining the active route is as follows:

1. Verify that the next hop can be resolved.
2. Choose the path with the lowest preference value (routing protocol process preference).  
Routes that are not eligible to be used for forwarding (for example, because they were rejected by routing policy or because a next hop is inaccessible) have a preference of -1 and are never chosen.
3. Prefer the path with higher local preference.  
For non-BGP paths, choose the path with the lowest **preference2** value.
4. If the accumulated interior gateway protocol (AIGP) attribute is enabled, prefer the path with the lower AIGP attribute.
5. Prefer the path with the shortest autonomous system (AS) path value (skipped if the **as-path-ignore** statement is configured).  
A confederation segment (sequence or set) has a path length of 0. An AS set has a path length of 1.
6. Prefer the route with the lower origin code.



Routes learned from an IGP have a lower origin code than those learned from an exterior gateway protocol (EGP), and both have lower origin codes than incomplete routes (routes whose origin is unknown).

7. Prefer the path with the lowest multiple exit discriminator (MED) metric.

Depending on whether nondeterministic routing table path selection behavior is configured, there are two possible cases:

- If nondeterministic routing table path selection behavior is not configured (that is, if the **path-selection cisco-nondeterministic** statement is not included in the BGP configuration), for paths with the same neighboring AS numbers at the front of the AS path, prefer the path with the lowest MED metric. To always compare MEDs whether or not the peer ASs of the compared routes are the same, include the **path-selection always-compare-med** statement.
- If nondeterministic routing table path selection behavior is configured (that is, the **path-selection cisco-nondeterministic** statement is included in the BGP configuration), prefer the path with the lowest MED metric.

Confederations are not considered when determining neighboring ASs. A missing MED metric is treated as if a MED were present but zero.

**NOTE:** MED comparison works for single path selection within an AS (when the route does not include an AS path), though this usage is uncommon.

By default, only the MEDs of routes that have the same peer autonomous systems (ASs) are compared. You can configure routing table path selection options to obtain different behaviors.

8. Prefer strictly internal paths, which include IGP routes and locally generated routes (static, direct, local, and so forth).
9. Prefer strictly external BGP (EBGP) paths over external paths learned through internal BGP (IBGP) sessions.
10. Prefer the path whose next hop is resolved through the IGP route with the lowest metric.

**NOTE:** A path is considered a BGP equal-cost path (and will be used for forwarding) if a tie-break is performed after the previous step. All paths with the same neighboring AS, learned by a multipath-enabled BGP neighbor, are considered.

BGP multipath does not apply to paths that share the same MED-plus-IGP cost yet differ in IGP cost. Multipath path selection is based on the IGP cost metric, even if two paths have the same MED-plus-IGP cost.

BGP compares the type of IGP metric before comparing the metric value itself in **rt\_metric2\_cmp**. For example, BGP routes that are resolved through IGP are preferred over discarded or rejected next-hops that are of type **RTM\_TYPE\_UNREACH**. Such routes are declared **inactive** because of their **metric-type**.

11. If both paths are external, prefer the currently active path to minimize route-flapping. This rule is not used if any one of the following conditions is true:
  - **path-selection external-router-id** is configured.
  - Both peers have the same router ID.
  - Either peer is a confederation peer.
  - Neither path is the current active path.
12. Prefer a primary route over a secondary route. A primary route is one that belongs to the routing table. A secondary route is one that is added to the routing table through an export policy.
13. Prefer the path from the peer with the lowest router ID. For any path with an originator ID attribute, substitute the originator ID for the router ID during router ID comparison.
14. Prefer the path with the shortest cluster list length. The length is 0 for no list.
15. Prefer the path from the peer with the lowest peer IP address.

## Routing Table Path Selection

The shortest AS path step of the algorithm, by default, evaluates the length of the AS path and determines the active path. You can configure an option that enables Junos OS to skip this step of the algorithm by including the **as-path-ignore** option.

**NOTE:** Starting with Junos OS Release 14.1R8, 14.2R7, 15.1R4, 15.1F6, and 16.1R1, the **as-path-ignore** option is supported for routing instances.

The routing process path selection takes place before BGP hands off the path to the routing table to make its decision. To configure routing table path selection behavior, include the **path-selection** statement:

```
path-selection {
  (always-compare-med | cisco-non-deterministic | external-router-id);
  as-path-ignore;
  l2vpn-use-bgp-rules;
  med-plus-igp {
    igp-multiplier number;
    med-multiplier number;
  }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Routing table path selection can be configured in one of the following ways:

- Emulate the Cisco IOS default behavior (**cisco-non-deterministic**). This mode evaluates routes in the order that they are received and does not group them according to their neighboring AS. With **cisco-non-deterministic** mode, the active path is always first. All inactive, but eligible, paths follow the active path and are maintained in the order in which they were received, with the most recent path first. Ineligible paths remain at the end of the list.

As an example, suppose you have three path advertisements for the 192.168.1.0 /24 route:

- Path 1—learned through EBGp; AS Path of 65010; MED of 200
- Path 2—learned through IBGP; AS Path of 65020; MED of 150; IGP cost of 5
- Path 3—learned through IBGP; AS Path of 65010; MED of 100; IGP cost of 10

These advertisements are received in quick succession, within a second, in the order listed. Path 3 is received most recently, so the routing device compares it against path 2, the next most recent advertisement. The cost to the IBGP peer is better for path 2, so the routing device eliminates path 3 from contention. When comparing paths 1 and 2, the routing device prefers path 1 because it is received from an EBGp peer. This allows the routing device to install path 1 as the active path for the route.

**NOTE:** We do not recommend using this configuration option in your network. It is provided solely for interoperability to allow all routing devices in the network to make consistent route selections.

- Always comparing MEDs whether or not the peer ASs of the compared routes are the same (**always-compare-med**).
- Override the rule that If both paths are external, the currently active path is preferred (**external-router-id**). Continue with the next step (Step 12) in the path-selection process.
- Adding the IGP cost to the next-hop destination to the MED value before comparing MED values for path selection (**med-plus-igp**).

BGP multipath does not apply to paths that share the same MED-plus-IGP cost, yet differ in IGP cost. Multipath path selection is based on the IGP cost metric, even if two paths have the same MED-plus-IGP cost.

## BGP Table path selection

The following parameters are followed for BGP's path selection:

1. Prefer the highest local-preference value.
2. Prefer the shortest AS-path length.
3. Prefer the lowest origin value.
4. Prefer the lowest MED value.
5. Prefer routes learned from an EBGp peer over an IBGP peer.
6. Prefer best exit from AS.
7. For EBGp-received routes, prefer the current active route.
8. Prefer routes from the peer with the lowest Router ID.
9. Prefer paths with the shortest cluster length.
10. Prefer routes from the peer with the lowest peer IP address. Steps 2, 6 and 12 are the RPD criteria.

## Effects of Advertising Multiple Paths to a Destination

BGP advertises only the active path, unless you configure BGP to advertise multiple paths to a destination.

Suppose a routing device has in its routing table four paths to a destination and is configured to advertise up to three paths (**add-path send path-count 3**). The three paths are chosen based on path selection criteria. That is, the three best paths are chosen in path-selection order. The best path is the active path. This path is removed from consideration and a new best path is chosen. This process is repeated until the specified number of paths is reached.

### Release History Table

Release	Description
<a href="#">14.1R8</a>	Starting with Junos OS Release 14.1R8, 14.2R7, 15.1R4, 15.1F6, and 16.1R1, the <b>as-path-ignore</b> option is supported for routing instances.

### RELATED DOCUMENTATION

*Example: Ignoring the AS Path Attribute When Selecting the Best Path*

*Examples: Configuring BGP MED*

*Example: Advertising Multiple BGP Paths to a Destination*

## Enabling BGP Path Selection for Layer 2 VPNs and VPLS

Layer 2 VPNs and VPLS share the same path selection process for determining the optimal path to reach all of the destinations shared within a single routing instance. For Layer 2 VPN and VPLS topologies, the path selection process is straightforward if there is just a single path from each PE router to each CE device. However, the path selection process becomes more complex if the PE routers receive two or more valid paths to reach a specific CE device.

**NOTE:** In the VPLS documentation, the word *router* in terms such as *PE router* is used to refer to any device that provides routing functions.

The following network scenarios provide examples of what might cause a PE router to receive more than one valid path to reach a specific CE device:

- **Multihoming**—One or more CE devices within a routing instance are multihomed to two or more PE routers. Each multihomed CE device has at least two valid paths.
- **Route reflectors**—There are multiple route reflectors deployed within the same network and they are supporting PE routers within the same routing instance. Due to time delays in large complex networks, the route reflectors can separately receive a different valid path to reach a CE device at different times. When they readvertise these valid paths, a PE router could receive two or more separate but apparently valid paths to the same CE device.

By default, Juniper Networks routers use just the designated forwarder path selection algorithm to select the best path to reach each Layer 2 VPN or VPLS routing instance destination (for more information, see *VPLS Path Selection Process for PE Routers*). However, you can also configure the routers in your network to use both the BGP path selection algorithm and the designated forwarder path selection algorithm as follows:

- On the Provider routers within the service providers network, the standard BGP path selection algorithm is used (for more information, see [“Understanding BGP Path Selection” on page 40](#)). Using the standard BGP path selection for Layer 2 VPN and VPLS routes allows a service provider to leverage the existing Layer 3 VPN network infrastructure to also support Layer 2 VPNs and VPLS. The BGP path selection algorithm also helps to ensure that the service provider’s network behaves predictably with regard to Layer 2 VPN and VPLS path selection. This is particularly important in networks employing route reflectors and multihoming.

When a Provider router receives multiple paths for the same destination prefix (for example, a multihomed CE device), one path is selected based on the BGP path selection algorithm and placed in the `bgp.l2vpn.0` routing table and the appropriate `instance.l2vpn.0` routing table.

- When a PE router receives all of the available paths to each CE device, it runs the designated forwarder path selection algorithm to select the preferred path to reach each CE device, independently of the results of the earlier BGP path selection algorithm run on the Provider router. The VPLS designated forwarder algorithm uses the D-bit, preference, and PE router identifier to determine which of the valid paths to each CE device to use. The PE router might select a path to reach a CE device which is different from the path selected by the BGP-based Provider routers. In this scenario, the following is the expected behavior for traffic sent to the multihomed CE device:
  - If the path selected by the remote PE router is available, traffic will traverse the network to the multihomed CE device using the remote PE router's preferred path (again, ignoring the path selected by the BGP-based Provider routers).
  - If the path selected by the remote PE router fails:
    1. The Provider routers switch the traffic destined for the multihomed CE device to the alternate path as soon as failure is detected.
    2. The Provider routers notify the remote PE routers of the path failure.
    3. The remote PE routers update their routing tables accordingly.

For more information about the VPLS designated forwarder path selection algorithm, see *VPLS Path Selection Process for PE Routers*. This algorithm is also described in the Internet draft draft-kompella-l2vpn-vpls-multihoming-03.txt, *Multi-homing in BGP-based Virtual Private LAN Service*.

To enable the BGP path selection algorithm for Layer 2 VPN and VPLS routing instances, complete the following steps:

1. Run Junos OS Release 12.3 or later on all of the PE and Provider routers participating in Layer 2 VPN or VPLS routing instances.
 

Attempting to enable this functionality on a network with a mix of routers that both do and do not support this feature can result in anomalous behavior.
2. Specify a unique route distinguisher on each PE router participating in a Layer 2 VPN or VPLS routing instance.
3. Configure the **l2vpn-use-bgp-rules** statement on all of the PE and Provider routers participating in Layer 2 VPN or VPLS routing instances.

You can configure this statement at the **[edit protocols bgp path-selection]** hierarchy level to apply this behavior to all of the routing instances on the router or at the **[edit routing-instances routing-instance-name protocols bgp path-selection]** hierarchy level to apply this behavior to a specific routing instance.

## RELATED DOCUMENTATION

[Understanding BGP Path Selection | 40](#)

---

*VPLS Path Selection Process for PE Routers*

---

[path-selection | 240](#)

---

[route-distinguisher | 136](#)



# Creating Backup Connections with Redundant Pseudowires

## IN THIS CHAPTER

- [Redundant Pseudowires for Layer 2 Circuits and VPLS | 49](#)
- [Configuring Redundant Pseudowires for Layer 2 Circuits and VPLS | 52](#)

## Redundant Pseudowires for Layer 2 Circuits and VPLS

## IN THIS SECTION

- [Types of Redundant Pseudowire Configurations | 50](#)
- [Pseudowire Failure Detection | 51](#)

A redundant pseudowire can act as a backup connection between PE routers and CE devices, maintaining Layer 2 circuit and VPLS services after certain types of failures. This feature can help improve the reliability of certain types of networks (metro for example) where a single point of failure could interrupt service for multiple customers. Redundant pseudowires cannot reduce traffic loss to zero. However, they provide a way to gracefully recover from pseudowire failures in such a way that service can be restarted within a known time limit.

**NOTE:** VPLS is not supported on ACX Series routers.

When you configure redundant pseudowires to remote PE routers, you configure one to act as the primary pseudowire over which customer traffic is being transmitted and you configure another pseudowire to act as a backup in the event the primary fails. You configure the two pseudowires statically. A separate label is allocated for the primary and backup neighbors.

For information about how to configure redundant pseudowires, see [“Configuring Redundant Pseudowires for Layer 2 Circuits and VPLS” on page 52](#).

The following sections provide an overview of redundant pseudowires for Layer 2 circuits and VPLS:

### Types of Redundant Pseudowire Configurations

You can configure redundant pseudowires for Layer 2 circuits and VPLS in either of the following manners:

**NOTE:** VPLS is not supported on ACX Series routers.

- You can configure a single active pseudowire. The PE router configured as the primary neighbor is given preference and this connection is the one used for customer traffic. For the LDP signalling, labels are exchanged for both incoming and outgoing traffic with the primary neighbor. The LDP label advertisement is accepted from the backup neighbor, but no label advertisement is forwarded to it, leaving the pseudowire in an incomplete state. The pseudowire to the backup neighbor is completed only when the primary neighbor fails. The decision to switch between the two pseudowires is made by the device configured with the redundant pseudowires. The primary remote PE router is unaware of the redundant configuration, ensuring that traffic is always switched using just the active pseudowire.
- Alternatively, you can configure two active pseudowires, one to each of the PE routers. Using this approach, control plane signalling is completed and active pseudowires are established with both the primary and backup neighbors. However, the data plane forwarding is done only over a one of the pseudowires (designated as the active pseudowire by the local device). The other pseudowire is on standby. The active pseudowire is preferably established with the primary neighbor and can switch to the backup pseudowire if the primary fails.

The decision to switch between the active and standby pseudowires is controlled by the local device. The remote PE routers are unaware of the redundant connection, and so both remote PE routers send traffic to the local device. The local device only accepts traffic from the active pseudowire and drops the traffic from the standby. In addition, the local device only sends traffic to the active pseudowire. If the active pseudowire fails, traffic is immediately switched to the standby pseudowire.

The two configurations available for pseudowire redundancy have the following limitations:

- For the single active pseudowire configuration, it takes more time (compared to the two active pseudowire configuration) to switchover to the backup pseudowire when a failure is detected. This approach requires additional control plane signalling to complete the pseudowire with the backup neighbor and traffic can be lost during the switchover from primary to backup.
- If you configure two active pseudowires, bandwidth is lost on the link carrying the backup pseudowire between the remote PE router and the local device. Traffic is always duplicated over both the active and standby pseudowires. The single active pseudowire configuration does not waste bandwidth in this fashion.

## Pseudowire Failure Detection

The following events are used to detect a failure (control and data plane) of the pseudowire configured between a local device and a remote PE router and initiates the switch to a redundant pseudowire:

- Manual switchover (user initiated)
- Remote PE router withdraws the label advertisement
- LSP to the remote PE router goes down
- LDP session with the remote PE router goes down
- Local configuration changes
- Periodic pseudowire OAM procedure fails (Layer 2 circuit-based MPLS ping to the PE router fails)

When you configure a redundant pseudowire between a CE device and a PE router, a periodic (once a minute) ping packet is forwarded through the active pseudowire to verify data plane connectivity. If the ping fails, traffic is automatically switched to the redundant pseudowire.

When a failure is detected, traffic is switched from the failed active pseudowire to the redundant pseudowire. The redundant pseudowire is then designated as the active pseudowire. The switch is nonreversible, meaning that once the redundant pseudowire assumes the role of the active pseudowire at the time of a failover, it remains as the active pseudowire even though the previously active pseudowire comes up again.

For example, a primary pseudowire has failed and traffic has been successfully switched to the redundant pseudowire. After a period of time, the cause of the failure of the primary pseudowire has been resolved and it is now possible to reestablish the original connection. However, traffic is not switched back to the original pseudowire unless a failure is detected on the currently active pseudowire.

## RELATED DOCUMENTATION

| *Example: Configuring H-VPLS Without VLANs*

## Configuring Redundant Pseudowires for Layer 2 Circuits and VPLS

### IN THIS SECTION

- [Configuring Pseudowire Redundancy on the PE Router | 52](#)
- [Configuring the Switchover Delay for the Pseudowires | 53](#)
- [Configuring a Revert Time for the Redundant Pseudowire | 53](#)

A redundant pseudowire can act as a backup connection between PE routers and CE devices, maintaining Layer 2 circuit and VPLS services after certain types of failures. This feature can help improve the reliability of certain types of networks (metro for example) where a single point of failure could interrupt service for multiple customers. Redundant pseudowires cannot reduce traffic loss to zero. However, they provide a way to gracefully recover from pseudowire failures in such a way that service can be restarted within a known time limit.

**NOTE:** VPLS is not supported on ACX Series routers.

For an overview of how redundant pseudowires work, see [“Redundant Pseudowires for Layer 2 Circuits and VPLS” on page 49](#).

To configure pseudowire redundancy for Layer 2 circuits and VPLS, complete the procedures in the following sections:

### Configuring Pseudowire Redundancy on the PE Router

You configure pseudowire redundancy on the PE router acting as the egress for the primary and standby pseudowires using the **backup-neighbor** statement.

To configure pseudowire redundancy on the PE router, include the **backup-neighbor** statement:

```
backup-neighbor {
```

```

community name;
psn-tunnel-endpoint address;
standby;
virtual-circuit-id number;
}

```

For a list of hierarchy levels at which you can include this statement, see the statement summary for this statement.

The **backup-neighbor** statement includes the following configuration options:

- **community**—Specifies the community for the backup neighbor.
- **psn-tunnel-endpoint**—Specifies the endpoint address for the packet switched network (PSN) tunnel on the remote PE router. The PSN tunnel endpoint address is the destination address for the LSP on the remote PE router.
- **standby**—Configures the pseudowire to the specified backup neighbor as the standby. When you configure this statement, traffic flows over both the active and standby pseudowires to the CE device. The CE device drops the traffic from the standby pseudowire, unless the active pseudowire fails. If the active pseudowire fails, the CE device automatically switches to the standby pseudowire.
- **virtual-circuit-id**—Uniquely identifies the primary and standby Layer 2 circuits. This option is configurable for Layer 2 circuits only.

## Configuring the Switchover Delay for the Pseudowires

To configure the time the router waits before switching traffic from the failed primary pseudowire to a backup pseudowire, include the **switchover-delay** statement:

```

switchover-delay milliseconds;

```

For a list of hierarchy levels at which you can include this statement, see the statement summary for this statement.

## Configuring a Revert Time for the Redundant Pseudowire

You can specify a revert time for redundant Layer 2 circuit and VPLS pseudowires. When you have configured redundant pseudowires for Layer 2 circuits or VPLS, traffic is switched to the backup pseudowire in the event that the primary pseudowire fails. If you configure a revert time, when the configured time expires traffic is reverted back to the primary pseudowire, assuming the primary pseudowire has been restored.

To configure a revert time for redundant pseudowires, specify the time in seconds using the **revert-time** statement:

```
revert-time (Protocols Layer 2 Circuits) seconds maximum seconds;
```

With the **maximum** option, specify a maximum reversion interval to add after the **revert-time** delay. If a revert-time delay is defined but a maximum timer is not defined, VCs are restored upon the revert-timer's expiration.

To reduce as much as possible the amount of traffic discarded, and potential data-path asymmetries observed during primary-to-backup transition periods, you can use this restoration timer. This restoration timer is activated when the backup path is performing as active, and then the primary path is restored. The goal is to avoid moving traffic back to the primary path right away, to make sure that the control plane's related tasks (such as IGP, LDP, RSVP, and internal BGP) have enough time to complete their updating cycle.

By enabling a gradual return of traffic to the primary path, you can ensure that the relatively-slow control-plane processing and updating does not have a negative impact on the restoration process.

The **maximum** option extends the revert timer's functionality to provide a jittered interval over which a certain number of circuits can be transitioned back to the primary path. By making use of this maximum value, you can define a time interval during which circuits are expected to switch over. As a consequence, circuits' effective transitions are scattered during restoration periods.

When making use of **revert-time x maximum y** statement, you can ensure that the corresponding circuit that is active is moved to the primary path within a time-slot ( $t_1$ ) such as that:  $x \leq t_1 \leq y$ . In other words, by activating this statement, you can ensure the following:

- VCs stay in the backup path for at least x seconds after the primary path comes back up.
- VCs are moved back to the primary path before y seconds have elapsed.
- $y \text{ maximum value} = x \text{ maximum value} * 2 = 1200 \text{ seconds}$ .

The ideal values for x and y will be conditioned to internal aspects of your network. For this reason, there are no default values for these settings. If no revert-time is set, the default behavior is non-revertive. That is, circuits are not returned to the primary path upon restoration. They are kept on the backup path.

For a list of hierarchy levels at which you can include this statement, see the statement summary for this statement.

## RELATED DOCUMENTATION

*Example: Configuring Pseudowire Redundancy in a Mobile Backhaul Scenario*

*Example: Configuring H-VPLS Without VLANs*

# Monitoring Layer 2 VPNs Using BFD

## IN THIS CHAPTER

- [Configuring BFD for Layer 2 VPN and VPLS | 56](#)
- [BFD Support for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS | 58](#)
- [Configuring BFD for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS | 59](#)

## Configuring BFD for Layer 2 VPN and VPLS

The following procedure describes how to configure Bidirectional Forwarding Detection (BFD) for Layer 2 VPN and VPLS. For VPNs, you configure the BFD sessions on the interfaces carrying traffic from the PE routers to the CE routers.

The BFD protocol is a simple hello mechanism that detects failures in a network. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. BFD works with a wide variety of network environments and topologies. The failure detection timers for BFD have shorter time limits than default failure detection mechanisms for BGP, so they provide faster detection.

The BFD failure detection timers are adaptive and can be adjusted to be faster or slower. The lower the BFD failure detection timer value, the faster the failure detection and vice versa. For example, the timers can adapt to a higher value if the adjacency fails (that is, the timer detects failures more slowly). Or a neighbor can negotiate a higher value for a timer than the configured value. The timers adapt to a higher value when a BFD session flap occurs more than three times in a span of 15 seconds. A back-off algorithm increases the receive interval by two if the local BFD instance is the reason for the session flap. The transmission interval is increased by two if the remote BFD instance is the reason for the session flap. You can use the **clear bfd adaptation** command to return BFD interval timers to their configured values. The **clear bfd adaptation** command is hitless, meaning that the command does not affect traffic flow on the routing device.



1. You can enable BFD failure detection. The BFD failure detection timers are adaptive and can be adjusted to be faster or slower. The lower the BFD failure detection timer value, the faster the failure detection and vice versa. For example, the timers can adapt to a higher value if the adjacency fails (that is, the timer detects failures more slowly). Or a neighbor can negotiate a higher value for a timer than the configured value. The timers adapt to a higher value when a BFD session flap occurs more than three times in a span of 15 seconds. A back-off algorithm increases the receive (Rx) interval by two if the local BFD instance is the reason for the session flap. The transmission (Tx) interval is increased by two if the remote BFD instance is the reason for the session flap.

To enable BFD failure detection and specify the threshold for the adaptation of the BFD session detection time, specify a time in milliseconds using the threshold statement. When the detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

**NOTE:** The threshold time must be equal to or greater than the value specified in the minimum-interval or the minimum-receive-interval statement.

You can use the **clear bfd adaptation** command to return BFD interval timers to their configured values. The **clear bfd adaptation** command is hitless, meaning that the command does not affect traffic flow on the routing device.

2. You can specify the minimum interval after which the local routing device transmits hello packets and then expects to receive a reply from a neighbor with which it has established a BFD session. You specify the interval in milliseconds using the **minimum-interval** statement.

Optionally, instead of using this statement, you can specify the minimum transmit and receive intervals separately using the minimum-interval (specified under the **transmit-interval** statement) and minimum-receive-interval statements.

3. You can configure the minimum interval after which the local routing device must receive a reply from a neighbor with which it has established a BFD session. Specify the number of milliseconds using the **minimum-receive-interval** statement.
4. You can specify that an interface be declared down when a certain number of hello packets have not been received from a neighboring router through that interface. Specify the number of hello packets by including the **multiplier** statement.
5. You can configure BFD sessions not to adapt to changing network conditions by including the **no-adaptation** statement. We recommend that you *do not* disable BFD adaptation unless it is preferable to have BFD adaptation disabled in your network.

6. Specify the transmit interval options for **bfd-liveness-detection** statement by including the **transmit-interval** statement. The negotiated transmit interval for a peer is the interval between the sending of BFD packets to peers. The receive interval for a peer is the minimum time that it requires between packets sent from its peer; the receive interval is not negotiated between peers. To determine the transmit interval, each peer compares its configured minimum transmit interval with its peer's minimum receive interval. The larger of the two numbers is accepted as the transmit interval for that peer.

The **transmit-interval** statement specifies how often BFD statements are transmitted and includes the following options:

- **minimum-interval milliseconds**—Specify the minimum interval in milliseconds at which the local routing device transmits hello packets to a neighbor with which it has established a BFD session.
- **threshold milliseconds**—Specify the threshold for the adaptation of the BFD session transmit interval. When the transmit interval adapts to a value greater than the threshold, a single trap and a single system message are sent.

**NOTE:** The threshold value specified in the **threshold** statement must be greater than the value specified in the **minimum-interval** statement for the **transmit-interval** statement.

7. Specify the BFD version by including the **version** statement. You can set BFD to version 1 or allow BFD to determine what version it needs to be by including the **automatic** option.

## RELATED DOCUMENTATION

[bfd-liveness-detection](#) | 157

*clear bfd adaptation*

## BFD Support for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS

Bidirectional Forwarding Detection (BFD) support for virtual circuit connectivity verification (VCCV) on MX Series devices enables you to configure a control channel for a pseudowire, in addition to the corresponding operations, administration, and management functions to be used over that control channel.

BFD provides a low resource mechanism for the continuous monitoring of the pseudowire data path and for detecting data plane failures. This feature provides support for asynchronous mode BFD for VCCV as described in RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*. Alternatively, you can use a ping operation to detect pseudowire failures. However,

the processing resources required for a ping operation are greater than what is needed for BFD. In addition, BFD is capable of detecting data plane failure faster than a VCCV ping. BFD for pseudowires is supported for Layer 2 circuits (LDP-based), Layer 2 VPNs (BGP-based), and VPLS (LDP-based or BGP-based).

Starting with Release 12.1, Junos OS introduces a distributed model for the BFD for VCCV. Unlike in previous releases where the BFD for VCCV followed a Routing Engine-based implementation, in Release 12.1 and later, the BFD for VCCV follows a distributed implementation over PIC concentrators, such as DPC, FPC, and MPC.

For distributed BFD, you need to configure the lo0 interface with unit 0 and the appropriate family enabled.

**NOTE:** For the distributed BFD for VCCV to work, you must configure MPLS family (**family mpls**) on the loopback interface.

```
user@router# set interfaces lo0 unit 0 family mpls
```

In Junos OS Release 12.1 and later, the periodic packet management process (ppmd) on the PIC concentrators handles the periodic packet management (send and receive) for BFD for VCCV. This enables Junos OS to create more BFD for VCCV sessions, and to reduce the time taken for error detection. Similarly, the distributed implementation improves the performance of Routing Engines because the Routing Engine resources used for BFD for VCCV implementation become available for Routing Engine-related applications when the BFD for VCCV-related processing moves to the PIC concentrators. The distributed BFD for VCCV implementation also enables the BFD for VCCV sessions to remain across graceful restarts.

## RELATED DOCUMENTATION

[Configuring BFD for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS | 59](#)

## Configuring BFD for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS

Bidirectional Forwarding Detection (BFD) support for virtual circuit connection verification (VCCV) allows you to configure a control channel for a pseudowire, in addition to the corresponding operations and management functions to be used over that control channel. BFD provides a low resource mechanism for the continuous monitoring of the pseudowire data path and for detecting data plane failures.

This feature provides support for asynchronous mode BFD for VCCV as described in RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*. You can also use a ping operation to detect pseudowire failures. However, the processing resources required for a ping

operation are greater than what is needed for BFD. In addition, BFD is capable of detecting data plane failure faster than VCCV ping. BFD for pseudowires is supported for Layer 2 circuits (LDP-based), Layer 2 VPNs (BGP-based), and VPLS (LDP-based or BGP-based).

To configure OAM and BFD for Layer 2 VPNs, include the **oam** statement and sub-statements at the **[edit routing-instances routing-instance-name protocols l2vpn]** hierarchy level:

```
oam {
  bfd-liveness-detection;
  ping-interval ;
  ping-multiplier;
}
```

You can configure many of the same OAM statements for VPLS and Layer 2 circuits:

- To enable OAM for VPLS, configure the **oam** statement and substatements at the **[edit routing-instances routing-instance-name protocols vpls]** hierarchy level and at the **[edit routing-instances routing-instance-name protocols vpls neighbor address]** hierarchy level. The **pwe3-control-word** statement configured at the **[edit routing-instances routing-instance-name protocols l2vpn oam control-channel]** hierarchy level is not applicable to VPLS configurations.
- To enable OAM for Layer 2 circuits, configure the **oam** statement and substatements at the **[edit protocols l2circuit neighbor address interface interface-name]** hierarchy level. The **control-channel** statement and sub-statements configured at the **[edit routing-instances routing-instance-name protocols l2vpn oam]** hierarchy level do not apply to Layer 2 circuit configurations.

You can use the **show ldp database extensive** command to display information about the VCCV control channel and the **show bfd session extensive** command to display information about BFD for Layer 2 VPNs, Layer 2 circuits, and VPLS.

## RELATED DOCUMENTATION

| Junos OS Routing Protocols Library

# 2

PART

## Configuring Layer 2 Circuits

---

[Overview | 62](#)

[Layer 2 Circuits Configuration Overview | 64](#)

[Configuring Protection Features for Layer 2 Circuits | 83](#)

[Monitoring Layer 2 Circuits with BFD | 102](#)

[Troubleshooting Layer 2 Circuits | 116](#)

---

# Overview

## IN THIS CHAPTER

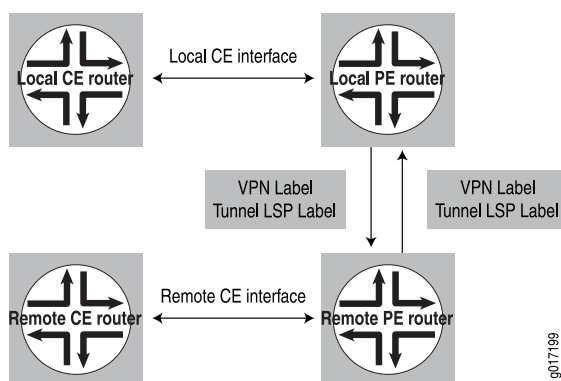
- [Layer 2 Circuit Overview](#) | 62

## Layer 2 Circuit Overview

A Layer 2 circuit is a point-to-point Layer 2 connection transported using Multiprotocol Label Switching (MPLS) or other tunneling technology on the service provider's network. A Layer 2 circuit is similar to a circuit cross-connect (CCC), except that multiple virtual circuits (VCs) are transported over a single shared label-switched path (LSP) tunnel between two provider edge (PE) routers. In contrast, each CCC requires a separate dedicated LSP.

The Junos OS implementation of Layer 2 circuits supports only the remote form of a Layer 2 circuit; that is, a connection from a local customer edge (CE) router to a remote CE router. [Figure 4 on page 62](#) illustrates the components of a Layer 2 circuit.

**Figure 4: Components of a Layer 2 Circuit**



To establish a Layer 2 circuit, the Label Distribution Protocol (LDP) is used as the signaling protocol to advertise the ingress label to the remote PE routers. For this purpose, a targeted remote LDP neighbor session is established using the extended discovery mechanism described in LDP, and the session is brought up to the remote PE loopback IP address. Because LDP looks at the Layer 2 circuit configuration and initiates extended neighbor discovery for all the Layer 2 circuit neighbors (the remote PEs), no new

configuration is necessary in LDP. Each Layer 2 circuit is represented by the logical interface connecting the local PE router to the local customer edge (CE) router. Note that LDP must be enabled on the lo0.0 interface for extended neighbor discovery to function correctly.

Packets are sent to remote CE routers over an egress VPN label advertised by the remote PE router, using a targeted LDP session. The VPN label is sent over an LDP LSP to the remote PE router connected to the remote CE router. Return traffic from the remote CE router destined to the local CE router is sent using an ingress VPN label advertised by the local PE router, which is also sent over the LDP LSP to the local PE router from the remote PE router.

## RELATED DOCUMENTATION

---

*Understanding Layer 3 VPNs*

---

[Layer 2 VPN Applications](#) | 3

---

*Applications for Interconnecting a Layer 2 Circuit with a Layer 2 Circuit*

---

*Applications for Interconnecting a Layer 2 Circuit with a Layer 3 VPN*

---

*Example: Interconnecting a Layer 2 Circuit with a Layer 2 Circuit*

---

*Example: Interconnecting a Layer 2 Circuit with a Layer 3 VPN*

---

*Example: Interconnecting a Layer 2 Circuit with a Layer 2 VPN*

# Layer 2 Circuits Configuration Overview

## IN THIS CHAPTER

- [Configuring Static Layer 2 Circuits | 64](#)
- [Configuring Local Interface Switching in Layer 2 Circuits | 65](#)
- [Configuring Interfaces for Layer 2 Circuits | 68](#)
- [Configuring Policies for Layer 2 Circuits | 78](#)
- [Configuring LDP for Layer 2 Circuits | 81](#)

## Configuring Static Layer 2 Circuits

You can configure static Layer 2 circuit pseudowires. Static pseudowires are designed for networks that do not support LDP or do not have LDP enabled. You configure a static pseudowire by configuring static values for the in and out labels needed to enable a pseudowire connection. The **ignore-mtu-mismatch**, **ignore-vlan-id**, and **ignore-encapsulation-mismatch** statements are not relevant for static pseudowire configurations since the peer router cannot forward this information.

When you configure static pseudowires, you need to manually compare the encapsulation, TDM bit rate, and control word of the router with the remote peer router and ensure that they match, otherwise the static pseudowire might not work.

To configure static Layer 2 circuit pseudowires, include the **static** statement:

```
static {  
    incoming-label label;  
    outgoing-label label;  
    send-oam;  
}
```

For a list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

You can configure a static pseudowire as a standalone Layer 2 circuit or in conjunction with a redundant pseudowire. You configure the static pseudowire statement at the **[edit protocols l2circuit neighbor address**



**interface** *interface-name*] hierarchy level. You configure the redundant pseudowire at the [edit protocols l2circuit neighbor *address* interface *interface-name* backup-neighbor *neighbor*] hierarchy level. If you configure a static pseudowire to a neighbor and also configure a redundant pseudowire, the redundant pseudowire must also be static.

You can enable the ability to ping a static pseudowire by configuring the **send-oam** statement. This functionality applies to the backup neighbor as well. Once you have configured this statement, you can ping the static pseudowire by issuing the **ping mpls l2circuit** command.

For information about how to configure redundant pseudowires, see [“Configuring Redundant Pseudowires for Layer 2 Circuits and VPLS” on page 52](#).

## RELATED DOCUMENTATION

[Configuring Redundant Pseudowires for Layer 2 Circuits and VPLS | 52](#)

[ping mpls l2circuit | 287](#)

## Configuring Local Interface Switching in Layer 2 Circuits

### IN THIS SECTION

- [Configuring the Interfaces for the Local Interface Switch | 66](#)
- [Enabling Local Interface Switching When the MTU Does Not Match | 67](#)

You can configure a virtual circuit entirely on the local router, terminating the circuit on a local interface. Possible uses for this feature include being able to enable switching between Frame Relay DLCIs.

To configure a virtual circuit to terminate locally, include the **local-switching** statement:

```
local-switching {
  interface interface-name {
    description text;
    end-interface {
      interface interface-name;
      no-revert;
      protect-interface interface-name;
    }
  }
}
```

```

ignore-mtu-mismatch;
no-revert;
protect-interface interface-name;
}
}

```

You can include this statement at the following hierarchy levels:

- [edit protocols l2circuit]
- [edit logical-systems *logical-system-name* protocols l2circuit]

**NOTE:** ACX Series routers do not support the [edit logical-systems] hierarchy level.

The following sections describe how to configure local interface switching:

### Configuring the Interfaces for the Local Interface Switch

Local interface switching requires you to configure at least two interfaces:

- Starting interface—Include the **interface** statement at the [edit protocols l2circuit local-switching] hierarchy level.
- Ending interface—Include the **end-interface** statement at the [edit protocols l2circuit local-switching interface *interface-name*] hierarchy level.

You can also configure virtual circuit interface protection for each local interface:

- Protect interface for the starting interface—Include the **protect-interface** statement at the [edit protocols l2circuit local-switching interface *interface-name*] hierarchy level.
- Protect interface for the ending interface—Include the **protect-interface** statement at the [edit protocols l2circuit local-switching interface *interface-name* end-interface] hierarchy level.

For more information about how to configure protect interfaces, see [“Configuring the Protect Interface” on page 73](#).

Typically, when the primary interface goes down, the pseudowire starts using the protect interface. By default, when the primary interface comes back online, the interface is switched-over back from the protect interface to the primary interface. To prevent the switchover back to the primary interface, unless the primary interface goes down, include the **no-revert** statement. This prevents loss of traffic during the switchover.

**NOTE:** If the protect interface fails, the interface is switched-over back to the primary interface, irrespective of whether or not the **no-revert** statement is included in the configuration.

You can configure the **no-revert** statement both for the starting interface and the ending interface.

```
[edit protocols l2circuit local-switching interface interface-name]
no-revert;
end-interface {
  interface interface-name;
  no-revert;
}
```

**NOTE:** The protect interface must be configured prior to configuring the **no-revert** statement.

### Enabling Local Interface Switching When the MTU Does Not Match

You can configure a local switching interface to ignore the MTU configuration set for the associated physical interface. This enables you to bring up a circuit between two logical interfaces that are defined on physical interfaces with different MTU values.

To configure the local switching interface to ignore the MTU configured for the physical interface, include the **ignore-mtu-mismatch** statement:

```
ignore-mtu-mismatch;
```

You can include this statement at the following hierarchy levels:

- [edit protocols l2circuit local-switching interface *interface-name*]
- [edit logical-systems *logical-system-name* protocols l2circuit local-switching interface *interface-name*]

**NOTE:** ACX Series routers do not support the [edit logical-systems] hierarchy level.

## Configuring Interfaces for Layer 2 Circuits

### IN THIS SECTION

- [Configuring the Address for the Neighbor of the Layer 2 Circuit | 68](#)
- [Configuring the Neighbor Interface for the Layer 2 Circuit | 69](#)
- [Configuring the Interface Encapsulation Type for Layer 2 Circuits | 76](#)
- [Configuring ATM2 IQ Interfaces for Layer 2 Circuits | 77](#)

The following sections describe how to configure interfaces for Layer 2 circuits:

**NOTE:** Not all subtasks are supported on all platforms; check the CLI on your device.

### Configuring the Address for the Neighbor of the Layer 2 Circuit

All the Layer 2 circuits using a particular remote PE router designated for remote CE routers are listed under the **neighbor** statement ("neighbor" designates the PE router). Each neighbor is identified by its IP address and is usually the end-point destination for the label-switched path (LSP) tunnel transporting the Layer 2 circuit.

To configure a PE router as a neighbor for a Layer 2 circuit, specify the neighbor address using the **neighbor** statement:

```
neighbor address {  
    ...  
}
```

You can include this statement at the following hierarchy levels:

- [edit protocols l2circuit]
- [edit logical-systems *logical-system-name* protocols l2circuit]

## Configuring the Neighbor Interface for the Layer 2 Circuit

### IN THIS SECTION

- [Configuring a Community for the Layer 2 Circuit | 70](#)
- [Configuring the Control Word for Layer 2 Circuits | 70](#)
- [Configuring the Encapsulation Type for the Layer 2 Circuit Neighbor Interface | 72](#)
- [Enabling the Layer 2 Circuit When the Encapsulation Does Not Match | 72](#)
- [Configuring the MTU Advertised for a Layer 2 Circuit | 73](#)
- [Enabling the Layer 2 Circuit When the MTU Does Not Match | 73](#)
- [Configuring the Protect Interface | 73](#)
- [Configuring the Protect Interface From Switching Over to the Primary Interface | 74](#)
- [Configuring the Pseudowire Status TLV | 74](#)
- [Configuring Layer 2 Circuits over Both RSVP and LDP LSPs | 75](#)
- [Configuring the Virtual Circuit ID | 76](#)

Each Layer 2 circuit is represented by the logical interface **encapsulation** connecting the local provider edge (PE) router to the local customer edge (CE) router. This interface is tied to the Layer 2 circuit neighbor configured in [“Configuring the Address for the Neighbor of the Layer 2 Circuit” on page 68](#).

To configure the interface for a Layer 2 circuit neighbor, include the **interface** statement:

**NOTE:** The commit operation fails, if the same logical interface is configured for both Layer 2 circuit and ccc connection.

**NOTE:** On the EX9200 switches, replace **encapsulation-type** with the **encapsulation** statement.

```
interface interface-name {
  bandwidth (bandwidth | ctnumber bandwidth);
  community community-name;
  (control-word | no-control-word);
  description text;
  encapsulation-type type;
```

```

ignore-encapsulation-mismatch;
ignore-mtu-mismatch;
mtu mtu-number;
no-revert;
protect-interface interface-name;
pseudowire-status-tlv;
psn-tunnel-endpoint address;
virtual-circuit-id identifier;
}

```

You can include this statement at the following hierarchy levels:

- [edit protocols l2circuit neighbor *address*]
- [edit logical-systems *logical-system-name* protocols l2circuit neighbor *address*]

The following sections describe how to configure the interface for the Layer 2 circuit neighbor:

#### **Configuring a Community for the Layer 2 Circuit**

To configure a community for a Layer 2 circuit, include the **community** statement:

```
community community-name;
```

You can include this statement at the following hierarchy levels:

- [edit protocols l2circuit neighbor *address* interface *interface-name*]
- [edit logical-systems *logical-system-name* protocols l2circuit neighbor *address* interface *interface-name*]

For information about how to configure a routing policy for a Layer 2 circuit, see [“Configuring Policies for Layer 2 Circuits” on page 78](#).

#### **Configuring the Control Word for Layer 2 Circuits**

##### **IN THIS SECTION**

- [Configuring the Control Word for Frame Relay Interfaces | 71](#)
- [Disabling the Control Word for Layer 2 Circuits | 72](#)

To emulate the virtual circuit (VC) encapsulation for Layer 2 circuits, a 4-byte control word is added between the Layer 2 protocol data unit (PDU) being transported and the VC label that is used for demultiplexing. For most protocols, a null control word consisting of all zeroes is sent between Layer 2 circuit neighbors.

However, individual bits are available in a control word that can carry Layer 2 protocol control information. The control information is mapped into the control word, which allows the header of a Layer 2 protocol to be stripped from the frame. The remaining data and control word can be sent over the Layer 2 circuit, and the frame can be reassembled with the proper control information at the egress point of the circuit.

The following Layer 2 protocols map Layer 2 control information into special bit fields in the control word:

- **Frame Relay**—The control word supports the transport of discard eligible (DE), forward explicit congestion notification (FECN), and backward explicit congestion notification (BECN) information. For configuration information, see [“Configuring the Control Word for Frame Relay Interfaces” on page 71](#).

**NOTE:** Frame Relay is not supported on the ACX Series routers.

- **ATM AAL5 mode**—The control word supports the transport of sequence number processing, ATM cell loss priority (CLP), and explicit forward congestion indication (EFCI) information. When you configure an AAL5 mode Layer 2 circuit, the control information is carried by default and no additional configuration is needed.
- **ATM cell-relay mode**—The control word supports sequence number processing only. When you configure a cell-relay mode Layer 2 circuit, the sequence number information is carried by default and no additional configuration is needed.

The Junos OS implementation of sequence number processing for ATM cell-relay mode and AAL5 mode is not the same as that described in Sec. 3.1.2 of the IETF draft *Encapsulation Methods for Transport of Layer 2 Frames Over IP and MPLS Networks*. The differences are as follows:

- A packet with a sequence number of 0 is considered as out of sequence.
- A packet that does not have the next incremental sequence number is considered out of sequence.
- When out-of-sequence packets arrive, the sequence number in the Layer 2 circuit control word increments by one and becomes the expected sequence number for the neighbor.

The following sections discuss how to configure the control word for Layer 2 circuits:

### ***Configuring the Control Word for Frame Relay Interfaces***

On interfaces with Frame Relay CCC encapsulation, you can configure Frame Relay control bit translation to support Frame Relay services over IP and MPLS backbones by using CCC, Layer 2 VPNs, and Layer 2 circuits. When you configure translation of Frame Relay control bits, the bits are mapped into the Layer 2 circuit control word and preserved across the IP or MPLS backbone.

For information about how to configure the control bits, see the *Configuring Frame Relay Control Bit Translation*.

### Disabling the Control Word for Layer 2 Circuits

The Junos OS can typically determine whether a neighboring router supports the control word. However, if you want to explicitly disable its use on a specific interface, include the **no-control-word** statement:

```
no-control-word;
```

For a list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

### Configuring the Encapsulation Type for the Layer 2 Circuit Neighbor Interface

You can specify the Layer 2 circuit encapsulation type for the interface receiving traffic from a Layer 2 circuit neighbor. The encapsulation type is carried in the LDP-signaling messages exchanged between Layer 2 circuit neighbors when pseudowires are created. The encapsulation type you configure for each Layer 2 circuit neighbor varies depending on the type of networking equipment or the type of Layer 2 protocol you have deployed in your network. If you do not specify an encapsulation type for the Layer 2 circuit, the encapsulation of the CE device interface is used by default.

Specify the encapsulation type for the Layer 2 circuit neighbor interface by including the **encapsulation-type** statement:

```
encapsulation-type (atm-aal5 | atm-cell | atm-cell-port-mode | atm-cell-vc-mode | atm-cell-vp-mode | cesop |  
cisco-hdlc | ethernet | ethernet-vlan | frame-relay | frame-relay-port-mode | interworking | ppp | satop-e1 |  
satop-e3 | satop-t1 | satop-t3);
```

You can include this statement at the following hierarchy levels:

- [edit protocols l2circuit neighbor *address* interface *interface-name*]
- [edit logical-systems *logical-system-name* protocols l2circuit neighbor *address* interface *interface-name*]

### Enabling the Layer 2 Circuit When the Encapsulation Does Not Match

You can configure the Junos OS to allow a Layer 2 circuit to be established even though the encapsulation configured on the CE device interface does not match the encapsulation configured on the Layer 2 circuit interface by including the **ignore-encapsulation-mismatch** statement. You can configure the **ignore-encapsulation-mismatch** statement for the connection to the remote connection by including the statement at the [edit protocols l2circuit neighbor *address* interface *interface-name*] hierarchy level or for the local connection by including this statement at the [edit protocols l2circuit local-switching interface *interface-name*] hierarchy level.

```
ignore-encapsulation-mismatch;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.



### Configuring the MTU Advertised for a Layer 2 Circuit

By default, the MTU used to advertise a Layer 2 circuit is determined by taking the interface MTU for the associated physical interface and subtracting the encapsulation overhead for sending IP packets based on the encapsulation.

However, encapsulations that support multiple logical interfaces (and multiple Layer 2 circuits) rely on the same interface MTU (since they are all associated with the same physical interface). This can prove to be a limitation for VLAN Layer 2 circuits using the same Ethernet interface or for Layer 2 circuit DLCIs using the same Frame Relay interface.

This can also affect multivendor environments. For example, if you have three PE devices supplied by different vendors and one of the devices only supports an MTU of 1500, even if the other devices support larger MTUs you must to configure the MTU as 1500 (the smallest MTU of the three PE devices).

You can explicitly configure which MTU is advertised for a Layer 2 circuit, even if the Layer 2 circuit is sharing a physical interface with other Layer 2 circuits. When you explicitly configure an MTU for a Layer 2 circuit, be aware of the following:

- An explicitly configured MTU is signaled to the remote PE device. The configured MTU is also compared to the MTU received from the remote PE device. If there is a conflict, the Layer 2 circuit is taken down.
- If you configure an MTU for an ATM cell relay interface on an ATM II PIC, the configured MTU is used to compute the cell bundle size advertised for that Layer 2 circuit, instead of the default interface MTU.
- A configured MTU is used only in the control plane. It is not enforced in the data plane. You need to ensure that the CE device for a given Layer 2 circuit uses the correct MTU for data transmission.

To configure the MTU for a Layer 2 circuit, include the **mtu** statement at the **[edit protocols l2circuit neighbor address interface interface-name]** hierarchy level.

```
mtu mtu-number;
```

### Enabling the Layer 2 Circuit When the MTU Does Not Match

You can configure the Junos OS to allow a Layer 2 circuit to be established even though the MTU configured on the PE router does not match the MTU configured on the remote PE router by including the **ignore-mtu-mismatch** statement at the **[edit protocols l2circuit neighbor address interface interface-name]** hierarchy level.

### Configuring the Protect Interface

You can configure a protect interface for the logical interface linking a virtual circuit to its destination, whether the destination is remote or local. A protect interface provides a backup for the protected interface in case of failure. Network traffic uses the primary interface only so long as the primary interface functions. If the primary interface fails, traffic is switched to the protect interface. The protect interface is optional.

To configure the protect interface, include the **protect-interface** statement:

```
protect-interface interface-name;
```

**NOTE:** The protect interface must be configured prior to configuring the **no-revert** statement.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For an example of how to configure a protect interface for a Layer 2 circuit, see *Example: Configuring Layer 2 Circuit Protect Interfaces*.

### **Configuring the Protect Interface From Switching Over to the Primary Interface**

Typically, when the primary interface goes down, the pseudowire starts using the protect interface. By default, when the primary interface comes back online, the interface is switched-over back from the protect interface to the primary interface. To prevent the switchover back to the primary interface, unless the protect interface goes down, include the **no-revert** statement. This prevents loss of traffic during the switchover.

**NOTE:** If the protect interface fails, the interface is switched-over back to the primary interface, irrespective of whether or not the **no-revert** statement is included in the configuration.

You can configure the **no-revert** statement at the **[edit protocols l2circuit neighbor address interface interface-name]** hierarchy level:

```
[edit protocols l2circuit neighbor address interface interface-name]
no-revert;
```

### **Configuring the Pseudowire Status TLV**

The pseudowire status type length variable (TLV) is used to communicate the status of a pseudowire back and forth between two PE routers. For Layer 2 circuit configurations, you can configure the PE router to negotiate the pseudowire with its neighbor using the pseudowire status TLV. This same functionality is also available for LDP VPLS neighbor configurations. The pseudowire status TLV is configurable for each pseudowire connection and is disabled by default. The pseudowire status negotiation process assures that a PE router reverts back to the label withdraw method for pseudowire status if its remote PE router neighbor does not support the pseudowire status TLV.

Unlike the control word, a PE router's ability to support the pseudowire status TLV is communicated when the initial label mapping message is sent to its remote PE router. Once the PE router transmits its support for the pseudowire status TLV to its remote PE router, it includes the pseudowire status TLV in every label mapping message sent to the remote PE router. If you disable support for the pseudowire status TLV on

the PE router, a label withdraw message is sent to the remote PE router and then a new label mapping message without the pseudowire status TLV follows.

To configure the pseudowire status TLV for the pseudowire to the neighbor PE router, include the **pseudowire-status-tlv** statement:

```
pseudowire-status-tlv;
```

For a list of the hierarchy levels at which you can include this statement, see the statement summary section for this statement.

### **Configuring Layer 2 Circuits over Both RSVP and LDP LSPs**

You can configure two Layer 2 circuits between the same two routers, and have one Layer 2 circuit traverse an RSVP LSP and the other traverse an LDP LSP. To accomplish this, you need to configure two loopback addresses on the local router. You configure one of the loopback address for the Layer 2 circuit traversing the RSVP LSP. You configure the other loopback address to handle the Layer 2 circuit traversing the LDP LSP. For information about how to configure multiple loop back interfaces, see *Configuring Logical Units on the Loopback Interface for Routing Instances in Layer 3 VPNs*.

You also need to configure a packet switched network (PSN) tunnel endpoint for one of the Layer 2 circuits. It can be either the Layer 2 circuit traversing the RSVP LSP or the one traversing the LDP LSP. The PSN tunnel endpoint address is the destination address for the LSP on the remote router.

To configure the address for the PSN tunnel endpoint, include the **psn-tunnel-endpoint** statement:

```
psn-tunnel-endpoint address;
```

You can include this statement at the following hierarchy levels:

- [edit logical-systems *logical-system-name* protocols l2circuit neighbor *address* interface *interface-name*]
- [edit protocols l2circuit neighbor *address* interface *interface-name*]

By default, the PSN tunnel endpoint for a Layer 2 circuit is identical to the neighbor address, which is also the same as the LDP neighbor address.

The tunnel endpoints on the remote router do not need to be loopback addresses.

### **Example: PSN Tunnel Endpoint**

The following example illustrates how you might configure a PSN tunnel endpoint:

```
[edit protocols l2circuit]
neighbor 10.255.0.6 {
  interface t1-0/2/2.0 {
    psn-tunnel-endpoint 192.0.2.0;
```

```

        virtual-circuit-id 1;
    }
    interface t1-0/2/1.0 {
        virtual-circuit-id 10;
    }
}

```

The Layer 2 circuit configured for the **t1-0/2/2.0** interface resolves in the inet3 routing table to **192.0.2.0**. This could be either an RSVP route or a static route with an LSP next hop.

### Configuring the Virtual Circuit ID

You configure a virtual circuit ID on each interface. Each virtual circuit ID uniquely identifies the Layer 2 circuit among all the Layer 2 circuits to a specific neighbor. The key to identifying a particular Layer 2 circuit on a PE router is the neighbor address and the virtual circuit ID. An LDP-FEC-to-label binding is associated with a Layer 2 circuit based on the virtual circuit ID in the FEC and the neighbor that sent this binding. The LDP-FEC-to-label binding enables the dissemination of the VPN label used for sending traffic on that Layer 2 circuit to the remote CE device.

You also configure a virtual circuit ID for each redundant pseudowire. A redundant pseudowire is identified by the backup neighbor address and the virtual circuit ID. For more information, see [“Configuring Pseudowire Redundancy on the PE Router” on page 52](#).

To configure the virtual circuit ID, include the **virtual-circuit-id** statement:

```

virtual-circuit-id identifier;

```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

### Configuring the Interface Encapsulation Type for Layer 2 Circuits

The Layer 2 encapsulation type is carried in the LDP forwarding equivalence class (FEC). You can configure either circuit cross-connect (CCC) or translational cross-connect (TCC) encapsulation types for Layer 2 circuits. For more information, see the *MPLS Applications User Guide* and *Junos OS Network Interfaces Library for Routing Devices*.

**NOTE:** Some platform and FPC combinations can not pass TCC encapsulated ISO traffic. See *Platforms/FPCs That Cannot Forward TCC Encapsulated ISO Traffic* for details.

To configure the interface encapsulation for a Layer 2 circuit, include the **encapsulation** statement:

```
encapsulation encapsulation;
```

You can include this statement at the following hierarchy levels:

- [edit interfaces *interface-name*]
- [edit logical-systems *logical-system-name* interfaces *interface-name*]

## Configuring ATM2 IQ Interfaces for Layer 2 Circuits

You can configure Asynchronous Transfer Mode 2 (ATM2) intelligent queuing (IQ) interfaces for Layer 2 circuits by using Layer 2 circuit ATM Adaptation Layer 5 (AAL5) transport mode, Layer 2 circuit ATM cell relay mode, and the Layer 2 circuit ATM trunk mode.

The configuration statements are as follows:

- **atm-l2circuit-mode aal5**
- **atm-l2circuit-mode cell**
- **atm-l2circuit-mode trunk**

For more information about these statements, see the *Junos OS Administration Library*. For more information about how to configure ATM2 IQ interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

The Junos OS implementation of sequence number processing for Layer 2 circuit ATM cell relay mode and Layer 2 circuit AAL5 mode differs from that described in the Internet draft draft-martini-l2circuit-encap-mpls-11.txt, *Encapsulation Methods for Transport of Layer 2 Frames over MPLS Networks* (expires August 2006).

The Junos OS implementation has the following differences:

1. A packet with a sequence number of 0 is treated as out of sequence.
2. A packet that does not have the next incremental sequence number is considered out of sequence.

When out-of-sequence packets arrive, the expected sequence number for the neighbor is set to the sequence number in the Layer 2 circuit control word.

## Configuring Policies for Layer 2 Circuits

### IN THIS SECTION

- [Configuring the Layer 2 Circuit Community | 78](#)
- [Configuring the Policy Statement for the Layer 2 Circuit Community | 79](#)
- [Verifying the Layer 2 Circuit Policy Configuration | 81](#)

You can configure Junos routing policies to control the flow of packets over Layer 2 circuits. This capability allows you to provide different level of service over a set of equal-cost Layer 2 circuits. For example, you can configure a circuit for high-priority traffic, a circuit for average-priority traffic, and a circuit for low-priority traffic. By configuring Layer 2 circuit policies, you can ensure that higher-value traffic has a greater likelihood of reaching its destination.

The following sections explain how to configure Layer 2 circuit policies:

### Configuring the Layer 2 Circuit Community

To configure a community for Layer 2 circuits, include the **community** statement.

```
community community-name {
    members [ community-ids ];
}
```

You can include this statement at the following hierarchy levels:

- **[edit policy-options]**
- **[edit logical-systems *logical-system-name* policy-options]**

**name** identifies the community or communities.

**community-ids** identifies the type of community or extended community:

- A normal community uses the following community ID format:

**as-number:community-value**

**as-number** is the autonomous system (AS) number of the community member.

**community-value** is the identifier of the community member. It can be a number from 0 through 65,535.

- An extended community uses the following community ID format:

***type:administrator:assigned-number***

**type** is the type of target community. The target community identifies the route's destination.

**administrator** is either an AS number or an IP version 4 (IPv4) address prefix, depending on the type of community.

**assigned-number** identifies the local provider.

You also need to configure the community for the Layer 2 circuit interface; see [“Configuring a Community for the Layer 2 Circuit” on page 70](#).

## Configuring the Policy Statement for the Layer 2 Circuit Community

To configure a policy to send community traffic over a specific LSP, include the **policy-statement** statement:

```
policy-statement policy-name {
  term term-name {
    from community community-name;
    then {
      install-nexthop (except | lsp lsp-name | lsp-regex lsp-regular-expression);
      accept;
    }
  }
}
```

You can include this statement at the following hierarchy levels:

- [edit **policy-options**]
- [edit **logical-systems *logical-system-name* policy-options**]

To prevent the installation of any matching next hops, include the **install-nexthop** statement with the **except** option:

```
install-nexthop except;
```

You can include this statement at the following hierarchy levels:

- [edit **policy-options policy-statement *policy-name* term *term-name* then**]
- [edit **logical-systems *logical-system-name* policy-options policy-statement *policy-name* term *term-name* then**]

To assign traffic from a community to a specific LSP, include the **install-nexthop** statement with the **lsp *lsp-name*** option and the **accept** statement:

```
install-nexthop lsp lsp-name;
accept;
```

You can include these statements at the following hierarchy levels:

- [edit policy-options policy-statement *policy-name* term *term-name* then]
- [edit logical-systems *logical-system-name* policy-options policy-statement *policy-name* term *term-name* then]

You can also use a regular expression to select an LSP from a set of similarly named LSPs for the **install-nexthop** statement. To configure a regular expression, include the **install-nexthop** statement with the **lsp-regex** option and the **accept** statement:

```
install-nexthop lsp-regex lsp-regular-expression;
accept;
```

You can include these statements at the following hierarchy levels:

- [edit policy-options policy-statement *policy-name* term *term-name* then]
- [edit logical-systems *logical-system-name* policy-options policy-statement *policy-name* term *term-name* then]

#### **Example: Configuring a Policy for a Layer 2 Circuit Community**

The following example illustrates how you might configure a regular expression in a Layer 2 circuit policy. You create three LSPs to handle gold-tier traffic from a Layer 2 circuit. The LSPs are named **alpha-gold**, **beta-gold**, and **delta-gold**. You then include the **install-nexthop** statement with the **lsp-regex** option with the LSP regular expression **.\*-gold** at the [edit policy-options policy-statement *policy-name* term *term-name* then] hierarchy level:

```
[edit policy-options]
policy-statement gold-traffic {
  term to-gold-LSPs {
    from community gold;
    then {
      install-nexthop lsp-regex .*-gold;
      accept;
    }
  }
}
```

The community **gold** Layer 2 circuits can now use any of the **-gold** LSPs. Given equal utilization across the three **-gold** LSPs, LSP selection is made at random.



You need to apply the policy to the forwarding table. To apply a policy to the forwarding table, configure the **export** statement at the **[edit routing-options forwarding-table]** hierarchy level:

```
[edit routing-options forwarding-table]
export policy-name;
```

## Verifying the Layer 2 Circuit Policy Configuration

To verify that you have configured a policy for the Layer 2 circuit, issue the **show route table mpls detail** command. It should display the community for ingress routes that corresponds to the Layer 2 circuits, as shown by the following example:

```
user@host> show route table mpls detail
```

```
so-1/0/1.0 (1 entry, 1 announced)
*L2VPN Preference: 7
Next hop: via so-1/0/0.0 weight 1, selected
Label-switched-path to-community-gold
Label operation: Push 100000 Offset: -4
Next hop: via so-1/0/0.0 weight 1
Label-switched-path to-community-silver
Label operation: Push 100000 Offset: -4
Protocol next hop: 10.255.245.45
Push 100000 Offset: -4
Indirect next hop: 85333f0 314
State: <Active Int>
Local AS: 100
Age: 22
Task: Common L2 VC
Announcement bits (2): 0-KRT 1-Common L2 VC
AS path: I
Communities: 100:1
```

For more information about how to configure routing policies, see *Routing Policies, Firewall Filters, and Traffic Policers User Guide*.

## Configuring LDP for Layer 2 Circuits

Use LDP as the signaling protocol to advertise ingress labels to the remote PE routers. When configured, LDP examines the Layer 2 circuit configuration and initiates extended neighbor discovery for all the Layer 2

circuit neighbors (for example, remote PEs). This process is similar to how LDP works when tunneled over RSVP. You must run LDP on the **lo0.0** interface for extended neighbor discovery to function correctly.

For detailed information about how to configure LDP, see the *MPLS Applications User Guide*.

# Configuring Protection Features for Layer 2 Circuits

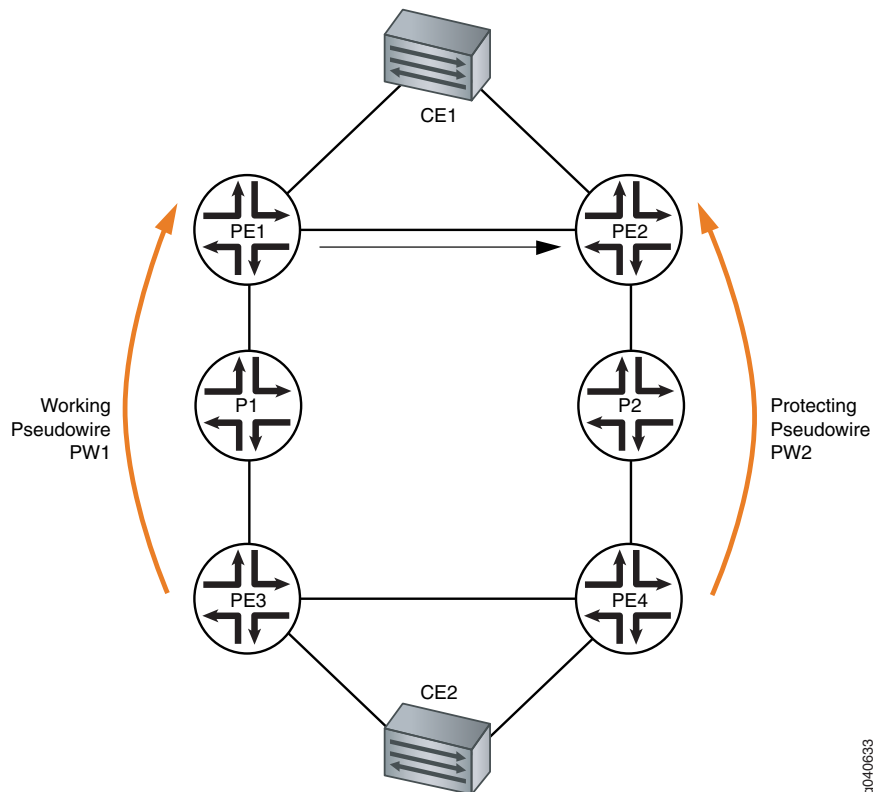
## IN THIS CHAPTER

- Egress Protection LSPs for Layer 2 Circuits | 83
- Example: Configuring Layer 2 Circuit Switching Protection | 84

## Egress Protection LSPs for Layer 2 Circuits

An egress protection LSP provides link protection for link between PE routers and CE devices as illustrated in [Figure 5 on page 83](#).

Figure 5: Egress Protection LSP



Device CE1 is multihomed to router PE1 and router PE2. Device CE2 is multihomed to router PE3 and router PE4. There are two paths connecting devices CE1 and CE2. The working path is CE2-PE3-P1-PE1-CE1, using pseudowire PW1. The protecting path is CE2-PE4-P2-PE2-CE1, using pseudowire PW2. Normally, traffic flows through the working path. When the end-to-end OAM between devices CE1 and CE2 detects a failure on the working path, traffic will be switched from the working path to the protecting path.

In the topology shown in [Figure 5 on page 83](#), if there was a link or node failure in the core network (for example, a link failure from router P1 to PE1, from router PE3 to P1, or a node failure of router P1), MPLS fast reroute can be triggered on the transport LSPs between router PE3 and router PE1 to repair the connection within tens of milliseconds. Egress protection LSPs address the problem of when a link failure occurs at the edge of the network (for example, a link failure on router PE1 to device CE1).

An egress protection LSP has been configured from router PE1 to router PE2. In the event of a link failure between router PE1 and device CE1, traffic can be switched to the egress protection LSP. Traffic from device CE2 can now be routed through path PE3-P1-PE1-PE2 to reach device CE1.

## Example: Configuring Layer 2 Circuit Switching Protection

### IN THIS SECTION

- [Requirements | 85](#)
- [Overview | 85](#)
- [Configuration | 86](#)

Unlike Layer 2 circuit protect interfaces (see *Example: Configuring Layer 2 Circuit Protect Interfaces*), which provide traffic protection for paths configured between the PE routers and CE routers, Layer 2 circuit switching protection provides traffic protection for the paths configured between the PE routers. In the event the path used by a Layer 2 circuit fails, traffic can be switched to an alternate path (or protection path). Switching protection is supported for locally switched Layer 2 circuits and provides 1 to 1 protection for each Layer 2 circuit interface.

When you enable Layer 2 circuit switching protection, each Layer 2 circuit interface requires the following paths:

- Working path—Used by the Layer 2 circuit when working normally.
- Protection path—Used by the Layer 2 circuit when the working path fails.

## Requirements

This example uses the following hardware and software components:

- MX Series 5G Universal Routing Platforms
- Junos OS Release 12.3

## Overview

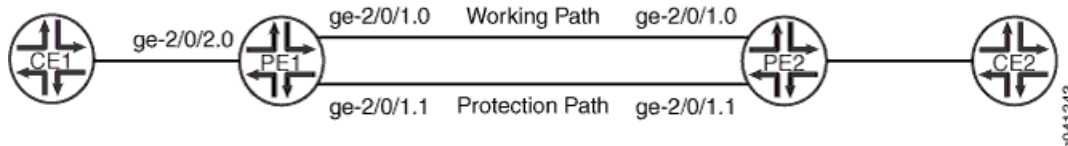
Each working path can be configured to have either a protection path routed directly to the neighboring PE router (as shown in [Figure 6 on page 85](#)) or indirectly using a pseudowire configured through an intermediate PE router (as shown in [Figure 7 on page 86](#) and [Figure 8 on page 86](#)). The protection path provides failure protection for the traffic flowing between the PE routers. Ethernet OAM monitors the status of these paths. When OAM detects a failure, it reroutes the traffic from the failed working path to the protection path. You can configure OAM to revert the traffic automatically to the working path when it is restored. You can also manually switch traffic between the working path, the protection path, and back.

**NOTE:** Non-stop routing (NSR) and graceful routing engine switchover (GRES) do not support Layer 2 circuit switching protection.

## Topology

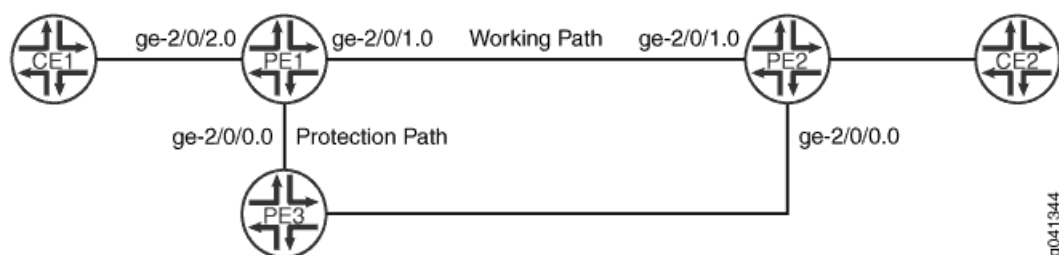
[Figure 6 on page 85](#) illustrates Layer 2 circuit local switching. There are two OAM sessions running between Router PE1 and Router PE2. One OAM session is configured over the working path and the other is configured over the protection path.

Figure 6: Connection Protection Enabled Between Router PE1 and Router PE2



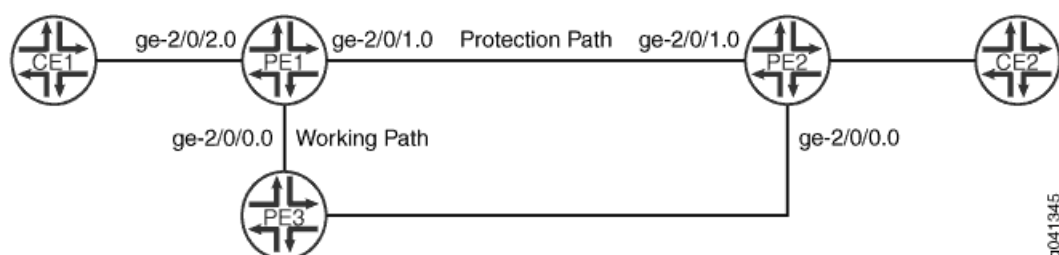
In [Figure 7 on page 86](#) and [Figure 8 on page 86](#), there are two OAM sessions running between Router PE1 and Router PE2. For Figure 2, one OAM session is configured over the working path between Router PE1 and Router PE2. The other OAM session is configured over the protection path between Router PE1 and Router PE3 to Router PE2.

Figure 7: Connection Protection Using a Pseudowire Configured through Router PE3 as the Protection Path



For [Figure 8 on page 86](#), one OAM session is configured over the working path, the pseudowire between Router PE1 and Router PE3, then to Router PE2. The other OAM session is configured on the protect path between Router PE1 and Router PE2.

Figure 8: Connection Protection Using a Pseudowire Configured through Router PE3 as the Working Path



## Configuration

### IN THIS SECTION

- [Configuring Connection Protection Between Two PE Routers | 87](#)
- [Verifying that OAM CFM Connections are Active | 91](#)
- [Configuring Connection Protection Using Another PE Router for the Protection Path | 92](#)
- [Verifying that OAM CFM Connections are Active | 96](#)
- [Configuring Connection Protection Using an Another PE Router for the Working Path | 96](#)
- [Verifying that OAM CFM Connections are Active | 100](#)

The following sections describe how to configure each of the variations of Layer 2 circuit connection protection:

## Configuring Connection Protection Between Two PE Routers

### Step-by-Step Procedure

To configure Layer 2 Circuit switching protection as shown in [Figure 6 on page 85](#) on Router PE1:

1. Configure the Layer 2 circuit on Router PE1.

```
[edit protocols l2circuit]
user@PE1# set local-switching interface ge-2/0/2.0 connection-protection
user@PE1# set local-switching interface ge-2/0/2.0 end-interface interface ge-2/0/1.0
user@PE1# set local-switching interface ge-2/0/2.0 end-interface backup-interface ge-2/0/1.1
```

2. Configure the routing policy on Router PE1.

```
[edit policy-options]
user@PE1# set policy-statement protection-policy then load-balance per-packet
```

3. Enable the routing policy on Router PE1.

```
[edit routing-options]
user@PE1# set forwarding-table export protection-policy
```

4. Configure OAM on Router PE1. OAM is used to monitor the working path between Router PE1 and Router PE2. In the event of a failure on the working path, traffic is switched automatically to the protection path. A connectivity fault management (CFM) session is configured on the working path and on the protection path. Begin by configuring the OAM maintenance domain.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md level 5
```

5. Configure OAM on Router PE1 for the working path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
  maintenance-association working continuity-check interval 100ms
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
  maintenance-association working mep 1000 interface ge-2/0/1.0
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
  maintenance-association working mep 1000 interface working
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
  maintenance-association working mep 1000 direction down
```

```
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 1000 remote-mep 103
```

6. Configure OAM on Router PE1 for the protection path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection continuity-check interval 100ms
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 interface ge-2/0/1.1
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 interface protect
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 direction down
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 remote-mep 104
```

7. Configure the OAM maintenance domain on Router PE2.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md level 5
```

8. Configure OAM on Router PE2 for the working path.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working continuity-check interval 100ms
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 interface ge-2/0/1.0
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 interface working
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 direction down
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 remote-mep 1000
```

9. Configure OAM on Router PE2 for the protection path.

```
[edit protocols oam ethernet]
```



```

user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection continuity-check interval 100ms
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 interface ge-2/0/1.1
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 interface protect
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 direction down
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 remote-mep 1001

```

## Results

From configuration mode on Router PE1, confirm your configuration by entering the **show protocols l2circuit**, **show policy-options**, **show routing-options**, and **show protocols oam ethernet** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```

user@host> show protocols l2circuit
local-switching {
  interface ge-2/0/2.0 {
    connection-protection;
    end-interface {
      interface ge-2/0/1.0;
      backup-interface ge-2/0/1.1;
    }
  }
}

```

```

user@host> show policy-options
policy-statement protection-policy {
  then {
    load-balance per-packet;
  }
}

```

```

user@host> show routing-options
forwarding-table {
  export protection-policy;
}

```

```

user@host> show protocols oam ethernet
connectivity-fault-management {
  maintenance-domain l2circuit-example-md {
    level 5;
    maintenance-association working {
      continuity-check {
        interval 100ms;
      }
      mep 1000 {
        interface ge-2/0/1.0 working;
        direction down;
        remote-mep 103;
      }
    }
  }
  maintenance-association protection {
    continuity-check {
      interval 100ms;
    }
    mep 1001 {
      interface ge-2/0/1.1 protect;
      direction down;
      remote-mep 104;
    }
  }
}
}

```

From configuration mode on Router PE2, confirm your configuration by entering the **show protocols oam ethernet** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```

connectivity-fault-management {
  maintenance-domain l2circuit-example-md {
    level 5;
    maintenance-association working {
      continuity-check {
        interval 100ms;
      }
      mep 103 {
        interface ge-2/0/1.0 working;
        direction down;
        remote-mep 1000;
      }
    }
  }
}

```

```

maintenance-association protection {
  continuity-check {
    interval 100ms;
  }
  mep 104 {
    interface ge-2/0/1.1 protect;
    direction down;
    remote-mep 1001;
  }
}
}
}

```

### **Verifying that OAM CFM Connections are Active**

#### **Purpose**

Verify that the CFM connections are active on each of the PE routers.

#### **Action**

Execute the following command on each of the PE routers.

1. Verify that the CFM working connection on Router PE1 is active.

```

user@ PE1> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association working

```

```

Interface status: Active, Link status: Up

```

2. Verify that the CFM protect connection on Router PE1 is active

```

user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association protection

```

```

Interface status: Active, Link status: Up

```

3. Verify that the CFM working connection on Router PE2 is active.

```

user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association working

```

```

Interface status: Active, Link status: Up

```

4. Verify that the CFM protect connection on Router PE2 is active.

```
user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association protection
```

```
Interface status: Active, Link status: Up
```

### *Configuring Connection Protection Using Another PE Router for the Protection Path*

#### **Step-by-Step Procedure**

To configure Layer 2 Circuit switching protection as shown in [Figure 7 on page 86](#) on Router PE1:

1. Configure the Layer 2 circuit on Router PE1.

```
[edit protocols l2circuit]
user@PE1# set local-switching interface ge-2/0/2.0 connection-protection
user@PE1# set local-switching interface ge-2/0/2.0 backup-neighbor 192.0.2.2 virtual-circuit-id 2
user@PE1# set local-switching interface ge-2/0/2.0 backup-neighbor 192.0.2.2 community example
user@PE1# set local-switching interface ge-2/0/2.0 end-interface interface ge-2/0/1.0
```

2. Configure the routing policy on Router PE1.

```
[edit policy-options]
user@PE1# set policy-statement load-balance then load-balance per-packet
user@PE1# set policy-statement protection-policy term protect from community example
user@PE1# set policy-statement protection-policy term protect then install-nexthop lsp-regex lsp-protect-*
```

3. Configure the routing options on Router PE1.

```
[edit routing-options]
user@PE1# set forwarding-table export load-balance
```

4. Configure OAM on Router PE1 to setup the maintenance domain. OAM is used to monitor the working path between Router PE1 and Router PE2. In the event of a failure on the working path, traffic is switched automatically to the protection path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md level 5
```

5. Configure OAM on Router PE1 for the working path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 1000 interface ge-2/0/1.0
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 1000 direction down
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 1000 remote-mep 103
```

6. Configure OAM on Router PE1 for the protection path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 interface ge-2/0/0.0
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 direction down
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 remote-mep 104
```

7. Configure OAM on Router PE2 to setup the maintenance domain.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md level 5
```

8. Configure OAM on Router PE2 for the working path.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 interface ge-2/0/1.0
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 direction down
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 remote-mep 1000
```

9. Configure OAM on Router PE2 for the protection path.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 interface ge-2/0/0.0
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 direction down
```

```
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 remote-mep 1001
```

## Results

From configuration mode on Router PE1, confirm your configuration by entering the **show protocols l2circuit**, **show policy-options**, **show routing-options**, and **show protocols oam ethernet** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host> show protocols l2circuit
local-switching {
  interface ge-2/0/2.0 {
    connection-protection;
    backup-neighbor 192.0.2.2 {
      virtual-circuit-id 2;
      community example;
    }
  }
  end-interface {
    interface ge-2/0/1.0;
  }
}
}
```

```
user@host> show policy-options
policy-statement load-balance {
  then {
    load-balance per-packet;
  }
}
policy-statement protection-policy {
  term protect {
    from community example;
    then {
      install-nexthop lsp-regex lsp-protect-*;
    }
  }
}
}
```

```
user@host> show routing-options
forwarding-table {
  export load-balance;
}
```

```
}
```

```
user@host> show protocols oam ethernet
connectivity-fault-management {
  maintenance-domain l2circuit-example-md {
    level 5;
    maintenance-association working {
      mep 1000 {
        interface ge-2/0/1.0;
        direction down;
        remote-mep 103;
      }
    }
    maintenance-association protection {
      mep 1001 {
        interface ge-2/0/0.0;
        direction down;
        remote-mep 104;
      }
    }
  }
}
```

From configuration mode on Router PE2, confirm your configuration by entering the **show protocols oam ethernet** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
connectivity-fault-management {
  maintenance-domain l2circuit-example-md {
    level 5;
    maintenance-association working {
      mep 103 {
        interface ge-2/0/1.0;
        direction down;
        remote-mep 1000;
      }
    }
    maintenance-association protection {
      mep 104 {
        interface ge-2/0/0.0;
        direction down;
        remote-mep 1001;
      }
    }
  }
}
```

```

    }
  }
}

```

### **Verifying that OAM CFM Connections are Active**

#### **Purpose**

Verify that the CFM connections are active on each of the PE routers.

#### **Action**

Execute the following command on each of the PE routers.

1. Verify that the CFM working connection on Router PE1 is active.

```

user@ PE1> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association working

```

```

Interface status: Active, Link status: Up

```

2. Verify that the CFM protect connection on Router PE1 is active

```

user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association protection

```

```

Interface status: Active, Link status: Up

```

3. Verify that the CFM working connection on Router PE2 is active.

```

user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association working

```

```

Interface status: Active, Link status: Up

```

4. Verify that the CFM protect connection on Router PE2 is active.

```

user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain
l2circuit-example-md maintenance-association protection

```

```

Interface status: Active, Link status: Up

```

### **Configuring Connection Protection Using an Another PE Router for the Working Path**

#### **Step-by-Step Procedure**



To configure Layer 2 Circuit switching protection as shown in [Figure 8 on page 86](#) on Router PE1:

1. Configure the Layer 2 circuit on Router PE1.

```
[edit protocols l2circuit]
user@PE1# set neighbor 192.0.2.2 interface ge-2/0/2.0 virtual-circuit-id 2
user@PE1# set neighbor 192.0.2.2 interface ge-2/0/2.0 community example
user@PE1# set neighbor 192.0.2.2 interface ge-2/0/2.0 connection-protection
user@PE1# set neighbor 192.0.2.2 interface ge-2/0/2.0 backup-neighbor 192.0.2.3 virtual-circuit-id 3
user@PE1# set neighbor 192.0.2.2 interface ge-2/0/2.0 backup-neighbor 192.0.2.3 standby
```

2. Configure the policies on Router PE1.

```
[edit policy-options]
user@PE1# set policy-statement load-balance then load-balance per-packet
user@PE1# set policy-statement protection-policy term protect from community example
user@PE1# set policy-statement protection-policy term protect then install-nexthop lsp-regex lsp-primary
```

3. Configure the routing options on Router PE1.

```
[edit routing-options]
user@PE1# set forwarding-table export load-balance
```

4. Configure OAM on Router PE1 to setup the maintenance domain. OAM is used to monitor the working path between Router PE1 and Router PE2. In the event of a failure on the working path, traffic is switched automatically to the protection path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md level 5
```

5. Configure OAM on Router PE1 for the working path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
  maintenance-association working mep 1000 interface ge-2/0/0.0
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
  maintenance-association working mep 1000 direction down
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
  maintenance-association working mep 1000 remote-mep 103
```

6. Configure OAM on Router PE1 for the protection path.

```
[edit protocols oam ethernet]
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 interface ge-2/0/1.0
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 direction down
user@PE1# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 1001 remote-mep 104
```

7. Configure OAM on Router PE2 to setup the maintenance domain.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md level 5
```

8. Configure OAM on Router PE2 for the working path.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 interface ge-2/0/0.0
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 direction down
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association working mep 103 remote-mep 1000
```

9. Configure OAM on Router PE2 for the protection path.

```
[edit protocols oam ethernet]
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 interface ge-2/0/1.0
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 direction down
user@PE2# set connectivity-fault-management maintenance-domain l2circuit-example-md
maintenance-association protection mep 104 remote-mep 1001
```

## Results

From configuration mode on Router PE1, confirm your configuration by entering the **show protocols l2circuit**, **show policy-options**, **show routing-options**, and **show protocols oam ethernet** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host> show protocols l2circuit
```

```
neighbor 192.0.2.2 {
  interface ge-2/0/2.0 {
    virtual-circuit-id 2;
    community example;
    connection-protection;
    backup-neighbor 192.0.2.3 {
      virtual-circuit-id 3;
      standby;
    }
  }
}
```

```
user@host> show policy-options
```

```
policy-statement load-balance {
  then {
    load-balance per-packet;
  }
}
policy-statement protection-policy {
  term protect {
    from community example;
    then {
      install-nexthop lsp-regex lsp-primary;
    }
  }
}
```

```
user@host> show routing-options
```

```
forwarding-table {
  export load-balance;
}
```

```
user@host> show protocols oam ethernet
```

```
connectivity-fault-management {
  maintenance-domain l2circuit-example-md {
    level 5;
    maintenance-association working {
      mep 1000 {
        interface ge-2/0/0.0;
        direction down;
        remote-mep 103;
      }
    }
  }
}
```

```

    }
  }
  maintenance-association protection {
    mep 1001 {
      interface ge-2/0/1.0;
      direction down;
      remote-mep 104;
    }
  }
}

```

From configuration mode on Router PE2, confirm your configuration by entering the **show protocols oam ethernet** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```

connectivity-fault-management {
  maintenance-domain l2circuit-example-md {
    level 5;
    maintenance-association working {
      mep 103 {
        interface ge-2/0/0.0;
        direction down;
        remote-mep 1000;
      }
    }
    maintenance-association protection {
      mep 104 {
        interface ge-2/0/1.0;
        direction down;
        remote-mep 1001;
      }
    }
  }
}

```

### ***Verifying that OAM CFM Connections are Active***

#### **Purpose**

Verify that the CFM connections are active on each of the PE routers.

#### **Action**

Execute the following command on each of the PE routers.

1. Verify that the CFM working connection on Router PE1 is active.

```
user@ PE1> show oam ethernet connectivity-fault-management mep-database maintenance-domain  
l2circuit-example-md maintenance-association working
```

```
Interface status: Active, Link status: Up
```

2. Verify that the CFM protect connection on Router PE1 is active

```
user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain  
l2circuit-example-md maintenance-association protection
```

```
Interface status: Active, Link status: Up
```

3. Verify that the CFM working connection on Router PE2 is active.

```
user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain  
l2circuit-example-md maintenance-association working
```

```
Interface status: Active, Link status: Up
```

4. Verify that the CFM protect connection on Router PE2 is active.

```
user@ PE2> show oam ethernet connectivity-fault-management mep-database maintenance-domain  
l2circuit-example-md maintenance-association protection
```

```
Interface status: Active, Link status: Up
```

## RELATED DOCUMENTATION

| *Example: Configuring Layer 2 Circuit Protect Interfaces*

# Monitoring Layer 2 Circuits with BFD

## IN THIS CHAPTER

- [Configuring BFD for VCCV for Layer 2 Circuits | 102](#)
- [Example: Configuring BFD for VCCV for Layer 2 Circuits | 105](#)

## Configuring BFD for VCCV for Layer 2 Circuits

Bidirectional Forwarding Detection (BFD) support for virtual circuit connection verification (VCCV) allows you to configure a control channel for a pseudowire, in addition to the corresponding operations and management functions to be used over that control channel. BFD provides a low resource mechanism for the continuous monitoring of the pseudowire data path and for detecting data plane failures. This feature provides support for asynchronous mode BFD for VCCV as described in RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*. You can also use a ping to detect pseudowire failures. However, the processing resources required for a ping are greater than what is needed for BFD. In addition, BFD is capable of detecting data plane failure faster than VCCV ping. BFD for pseudowires is supported for Layer 2 circuits (LDP-based).

Before you begin:

- Configure the device interfaces.

To configure BFD for VCCV:

1. Specify the threshold for the adaptation of the BFD session detection time.

```
[edit protocols l2circuit neighbor IP-address interface interface-name oam bfd-liveness-detection]
user@host# set detection-time threshold milliseconds
```

For example, to set a detection time threshold of 40 milliseconds for OAM BFD liveness detection:

```
[edit protocols l2circuit neighbor 192.0.2.1 interface ge-1/1/9.0 oam bfd-liveness-detection]
user@host# set detection-time threshold 40
```

2. Configure the virtual circuit ID for the Layer 2 circuit protocol.

```
[edit protocols l2circuit neighbor IP-address interface interface-name]
user@host# set virtual-circuit-id virtual-circuit-id
```

For example, to set the virtual circuit ID as 1 for OAM BFD liveness detection:

```
[edit protocols l2circuit neighbor 192.0.2.1 interface ge-1/1/9.0 oam bfd-liveness-detection]
user@host# set virtual-circuit-id 1
```

3. Configure the minimum interval after which the local routing device transmits hello packets and then expects to receive a reply from a neighbor with which it has established a BFD session for the Layer 2 circuit.

```
[edit protocols l2circuit neighbor IP-address interface interface-name oam bfd-liveness-detection]
user@host# set minimum-interval milliseconds
```

For example, to set a minimum interval of 300 milliseconds for OAM BFD liveness detection:

```
[edit protocols l2circuit neighbor 192.0.2.1 interface ge-1/1/9.0 oam bfd-liveness-detection]
user@host# set minimum-interval 300
```

4. Configure the minimum interval after which the local routing device must receive a reply from a neighbor with which it has established a BFD session for the Layer 2 circuit protocol.

```
[edit protocols l2circuit neighbor IP-address interface interface-name oam bfd-liveness-detection]
user@host# set minimum-receive-interval milliseconds
```

For example, to set a minimum receive interval of 10 milliseconds for OAM BFD liveness detection:

```
[edit protocols l2circuit neighbor 192.0.2.1 interface ge-1/1/9.0 oam bfd-liveness-detection]
user@host# set minimum-receive-interval 10
```

5. Configure the number of hello packets not received by a neighbor that causes the originating interface to be declared down for the Layer 2 circuit protocol.

```
[edit protocols l2circuit neighbor IP-address interface interface-name oam bfd-liveness-detection]
user@host# set multiplier number
```

For example, to set the multiplier as 3 for OAM BFD liveness detection:

```
[edit protocols l2circuit neighbor 192.0.2.1 interface ge-1/1/9.0 oam bfd-liveness-detection]
user@host# set multiplier 3
```

6. Configure to disable adaptation.

```
[edit protocols l2circuit neighbor IP-address interface interface-name oam bfd-liveness-detection]
user@host# set no-adaptation
```

7. Configure the minimum interval at which the local routing device transmits hello packets to a neighbor with which it has established a BFD session.

```
[edit protocols l2circuit neighbor IP-address interface interface-name oam bfd-liveness-detection
transmit-interval]
user@host# set minimum-interval milliseconds
```

For example, to set a minimum transmit interval of 5 milliseconds for OAM BFD liveness detection:

```
[edit protocols l2circuit neighbor 192.0.2.1 interface ge-1/1/9.0 oam bfd-liveness-detection transmit-interval]
user@host# set minimum-interval 5
```

8. Specify the threshold for the adaptation of the BFD session transmit interval.

```
[edit protocols l2circuit neighbor IP-address interface interface-name oam bfd-liveness-detection
transmit-interval]
user@host# set threshold milliseconds
```

For example, to set a transmit interval threshold of 30 milliseconds for OAM BFD liveness detection:

```
[edit protocols l2circuit neighbor 192.0.2.1 interface ge-1/1/9.0 oam bfd-liveness-detection transmit-interval]
user@host# set threshold 30
```

## RELATED DOCUMENTATION

[Example: Configuring BFD for VCCV for Layer 2 Circuits](#) | 105



## Example: Configuring BFD for VCCV for Layer 2 Circuits

### IN THIS SECTION

- [Requirements | 105](#)
- [Overview | 105](#)
- [Configuration | 106](#)
- [Verification | 112](#)

This example shows how to configure BFD for VCCV for Layer 2 circuits which enables faster detection of failure in the data path.

### Requirements

This example uses the following hardware and software components:

- Two MX Series 5G Universal Routing Platforms
- Junos OS Release 12.1 or later running on all devices

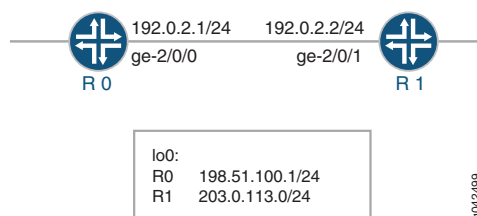
### Overview

Starting with Junos OS Release 12.1, Bidirectional Forwarding Detection (BFD) support for virtual circuit connection verification (VCCV) allows you to configure a control channel for a pseudowire, in addition to the corresponding operations and management functions to be used over that control channel. BFD provides a low resource mechanism for the continuous monitoring of the pseudowire data path and for detecting data plane failures. This feature provides support for asynchronous mode BFD for VCCV as described in RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*. You can also use a ping to detect pseudowire failures. However, the processing resources required for a ping are greater than what is needed for BFD. In addition, BFD is capable of detecting data plane failure faster than VCCV ping. BFD for pseudowires is supported for Layer 2 circuits (LDP-based).

To configure BFD for VCCV for Layer 2 circuits, configure the **oam** configuration statement at the **[edit protocols l2circuit neighbor address interface interface-name]** hierarchy level. The **control-channel** configuration statement at the **[edit routing-instances routing-instance-name protocols l2vpn oam]** hierarchy level does not apply to Layer 2 circuit configurations.

## Topology

In the topology, BFD for VCCV for Layer 2 circuits is configured on Device R0.



## 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.

R0

```

set chassis redundancy graceful-switchover
set interfaces ge-1/1/9 vlan-tagging
set interfaces ge-1/1/9 encapsulation vlan-ccc
set interfaces ge-1/1/9 unit 0 encapsulation vlan-ccc
set interfaces ge-1/1/9 unit 0 vlan-id 512
set interfaces ge-2/0/0 unit 0 family inet address 192.0.2.1/24
set interfaces ge-2/0/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 198.51.100.0/24
set routing-options nonstop-routing
set routing-options static route 203.0.113.0/24 next-hop 192.0.2.2
set routing-options router-id 198.51.100.0
set protocols rsvp interface ge-2/0/0.0
set protocols mpls label-switched-path lsp1 to 203.0.113.0
set protocols mpls interface ge-2/0/0.0
set protocols ospf traffic-engineering
set protocols ospf area 0.0.0.0 interface ge-2/0/0.0
set protocols ldp interface all
set protocols l2circuit neighbor 203.0.113.0 interface ge-1/1/9.0 virtual-circuit-id 1
set protocols l2circuit neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection
    minimum-interval 300

```

```

set protocols l2circuit neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  minimum-receive-interval 10
set protocols l2circuit neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection multiplier
  3
set protocols l2circuit neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  transmit-interval minimum-interval 5
set protocols l2circuit neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  transmit-interval threshold 30
set protocols l2circuit neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  detection-time threshold 40

```

R1

```

set interfaces ge-1/1/9 vlan-tagging
set interfaces ge-1/1/9 encapsulation vlan-ccc
set interfaces ge-1/1/9 unit 0 encapsulation vlan-ccc
set interfaces ge-1/1/9 unit 0 vlan-id 512
set interfaces ge-2/0/1 unit 0 family inet address 192.0.2.2/24
set interfaces ge-2/0/1 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 203.0.113.0/24
set routing-options static route 198.51.100.0/24 next-hop 192.0.2.1
set routing-options router-id 203.0.113.0
set protocols rsvp interface ge-2/0/1.0
set protocols mpls label-switched-path lsp2 to 198.51.100.0
set protocols mpls interface ge-2/0/1.0
set protocols ospf traffic-engineering
set protocols ospf area 0.0.0.0 interface ge-2/0/1.0
set protocols ldp interface all
set protocols l2circuit neighbor 198.51.100.0 interface ge-1/1/9.0 virtual-circuit-id 1
set protocols l2circuit neighbor 198.51.100.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  minimum-interval 300
set protocols l2circuit neighbor 198.51.100.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  minimum-receive-interval 10
set protocols l2circuit neighbor 198.51.100.0 interface ge-1/1/9.0 oam bfd-liveness-detection multiplier
  3
set protocols l2circuit neighbor 198.51.100.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  transmit-interval minimum-interval 5
set protocols l2circuit neighbor 198.51.100.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  transmit-interval threshold 30

```

```
set protocols l2circuit neighbor 198.51.100.0 interface ge-1/1/9.0 oam bfd-liveness-detection
detection-time threshold 40
```

## Configuring Device R0

### Step-by-Step Procedure

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

To configure Device R0:

**NOTE:** Repeat this procedure for Device R1 after modifying the appropriate interface names, addresses, and any other parameters for the device.

1. Configure graceful switchover redundancy.

```
[edit chassis]
user@R0# set redundancy graceful-switchover
```

2. Configure the interfaces.

```
[edit interfaces]
user@R0# set ge-1/1/9 vlan-tagging
user@R0# set ge-1/1/9 encapsulation vlan-ccc
user@R0# set ge-1/1/9 unit 0 encapsulation vlan-ccc
user@R0# set ge-1/1/9 unit 0 vlan-id 512

user@R0# set ge-2/0/0 unit 0 family inet address 192.0.2.1/24
user@R0# set ge-2/0/0 unit 0 family mpls

user@R0# set lo0 unit 0 family inet address 198.51.100.0/24
```

3. Configure the nonstop routing option, the static route, and the router ID routing options.

```
[edit routing-options]
user@R0# set nonstop-routing
user@R0# set static route 203.0.113.0/24 next-hop 192.0.2.2
```

```
user@R0# set router-id 198.51.100.0
```

4. Configure the RSVP protocol.

```
[edit protocols rsvp]
user@R0# set interface ge-2/0/0.0
```

5. Configure the MPLS protocol.

```
[edit protocols mpls]
user@R0# set label-switched-path lsp1 to 203.0.113.0
user@R0# set interface ge-2/0/0.0
```

6. Configure the OSPF protocol.

```
[edit protocols ospf]
user@R0# set traffic-engineering
user@R0# set area 0.0.0.0 interface ge-2/0/0.0
```

7. Configure the LDP protocol.

```
[edit protocols ldp]
user@R0# set interface all
```

8. Configure the virtual circuit ID for the neighbor of Layer 2 circuit protocols.

```
[edit protocols l2circuit]
user@R0# set neighbor 203.0.113.0 interface ge-1/1/9.0 virtual-circuit-id 1
```

9. Configure the oam attributes of the Layer 2 circuit protocol.

```
[edit protocols l2circuit]
user@R0# set neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection minimum-interval 300
user@R0# set neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection
  minimum-receive-interval 10
user@R0# set neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection multiplier 3
user@R0# set neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection transmit-interval
  minimum-interval 5
```

```

user@R0# set neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection transmit-interval
threshold 30
user@R0# set neighbor 203.0.113.0 interface ge-1/1/9.0 oam bfd-liveness-detection detection-time threshold
40

```

## Results

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

```

user@R0# show chassis
redundancy {
  graceful-switchover;
}

```

```

user@R0# show interfaces
ge-1/1/9 {
  vlan-tagging;
  encapsulation vlan-ccc;
  unit 0 {
    encapsulation vlan-ccc;
    vlan-id 512;
  }
}
ge-2/0/0 {
  unit 0 {
    family inet {
      address 192.0.2.1/24;
    }
    family mpls;
  }
}
lo0 {
  unit 0 {
    family inet {
      address 198.51.100.0/24;
    }
  }
}

```

```

user@R0# show protocols
rsvp {
    interface ge-2/0/0.0;
}
mpls {
    label-switched-path lsp1 {
        to 203.0.113.0;
    }
    interface ge-2/0/0.0;
}
ospf {
    traffic-engineering;
    area 0.0.0.0 {
        interface ge-2/0/0.0;
    }
}
ldp {
    interface all;
}
l2circuit {
    neighbor 203.0.113.0 {
        interface ge-1/1/9.0 {
            virtual-circuit-id 1;
            oam {
                bfd-liveness-detection {
                    minimum-interval 300;
                    minimum-receive-interval 10;
                    multiplier 3;
                    transmit-interval {
                        minimum-interval 5;
                        threshold 30;
                    }
                    detection-time {
                        threshold 40;
                    }
                }
            }
        }
    }
}
}

```

```

user@R0# show routing-options
nonstop-routing;
static {

```

```

route 203.0.113.0/24 next-hop 192.0.2.2;
}
router-id 198.51.100.0;

```

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

## Verification

### IN THIS SECTION

- [Verifying the Layer 2 Circuit Connections | 112](#)
- [Verifying the BFD Session | 113](#)
- [Verifying Detailed BFD Session Information | 114](#)

Verify that the configuration is working properly.

### *Verifying the Layer 2 Circuit Connections*

#### Purpose

Verify the connections in a Layer 2 Circuit.

#### Action

From operational mode, run the **show l2circuit connections** command for Device R0.

```
user@R0> show l2circuit connections
```

```
Layer-2 Circuit Connections:
```

```
Legend for connection status (St)
```

EI -- encapsulation invalid	NP -- interface h/w not present
MM -- mtu mismatch	Dn -- down
EM -- encapsulation mismatch	VC-Dn -- Virtual circuit Down
CM -- control-word mismatch	Up -- operational
VM -- vlan id mismatch	CF -- Call admission control failure
OL -- no outgoing label	IB -- TDM incompatible bitrate
NC -- intf encaps not CCC/TCC	TM -- TDM misconfiguration
BK -- Backup Connection	ST -- Standby Connection
CB -- rcvd cell-bundle size bad	SP -- Static Pseudowire



```

LD -- local site signaled down   RS -- remote site standby
RD -- remote site signaled down  HS -- Hot-standby Connection
XX -- unknown

Legend for interface status
Up -- operational
Dn -- down
Neighbor: 203.0.113.0
  Interface                Type  St    Time last up          # Up trans
ge-1/1/9.0(vc 1)          rmt   Up    Jun  2 03:19:44 2014      1
  Remote PE: 203.0.113.0, Negotiated control-word: Yes (Null)
  Incoming label: 299792, Outgoing label: 299792
  Negotiated PW status TLV: No
  Local interface: ge-1/1/9.0, Status: Up, Encapsulation: VLAN
  Flow Label Transmit: No, Flow Label Receive: No
  Flow Label Transmit: No, Flow Label Receive: No

```

### Meaning

The output shows the Layer 2 virtual circuit information from Device R0 to its neighbor.

### Verifying the BFD Session

#### Purpose

Verify the BFD session.

#### Action

From operational mode, run the **show bfd session** command for Device R0.

```
user@R0> show bfd session
```

```

          Detect   Transmit
Address      State  Interface  Time    Interval  Multiplier

203.0.113.7   Up    ge-2/0/0.0  0.030    0.010      3

1 sessions, 1 clients
Cumulative transmit rate 100.0 pps, cumulative receive rate 100.0 pps

```

### Meaning

The output shows the address, and the interface on which the BFD session is active. The state *Up* indicates that the BFD session is up. The BFD session has a time interval of 30 milliseconds to detect BFD control packets, the transmitting system has a time interval of 10 milliseconds to send BFD control packets, and the transmitting system determines the detection time by multiplying 3 with the time interval. Total number of active BFD sessions and total number of clients that are hosting active BFD sessions. Cumulative transmit rate indicates the total number of BFD control packets transmitted, per second, on all active sessions and cumulative receive rate indicates the total number of BFD control packets received, per second, on all active sessions.

### Verifying Detailed BFD Session Information

#### Purpose

Verify detailed BFD session information.

#### Action

From operational mode, run the **show bfd session extensive** command for Device R0.

```
user@R0> show bfd session extensive
```

Address	State	Interface	Detect Time	Transmit Interval	Multiplier
203.0.113.7	Up	ge-2/0/0.0	0.030	0.010	3

```

Client L2CKT-OAM, TX interval 0.005, RX interval 0.010
Session up time 03:47:14
Local diagnostic None, remote diagnostic None
Remote state Up, version 1
Replicated
Session type: VCCV BFD
Min async interval 0.005, min slow interval 1.000
Adaptive async TX interval 0.005, RX interval 0.010
Local min TX interval 0.005, minimum RX interval 0.010, multiplier 3
Remote min TX interval 0.005, min RX interval 0.010, multiplier 3
Threshold transmission interval 0.030, Threshold for detection time 0.040
Local discriminator 20, remote discriminator 13004
Echo mode disabled/inactive
Remote is control-plane independent
Neighbor address 203.0.113.0, Virtual circuit id 1
Session ID: 0x0

1 sessions, 1 clients
Cumulative transmit rate 100.0 pps, cumulative receive rate 100.0 pps

```

**Meaning**

The output shows detailed information for the BFD session.

**RELATED DOCUMENTATION**

| [Configuring BFD for VCCV for Layer 2 Circuits](#) | 102

# Troubleshooting Layer 2 Circuits

## IN THIS CHAPTER

- [Tracing Layer 2 Circuit Operations | 116](#)

## Tracing Layer 2 Circuit Operations

To trace the creation of and changes to Layer 2 circuits, include the **traceoptions** statement:

```
traceoptions {  
    file filename <files number> <size size> <world-readable | no-world-readable>;  
    flag flag <flag-modifier> <disable>;  
}
```

You can include this statement at the following hierarchy levels:

- **[edit protocols l2circuit]**
- **[edit logical-systems *logical-system-name* protocols l2circuit]**

Specify the following flags to trace the indicated operations on Layer 2 circuits:

- **connections**—Layer 2 circuit connections (events and state changes)
- **error**—Error conditions
- **FEC**—Layer 2 circuit advertisements received or sent using LDP
- **topology**—Layer 2 circuit topology changes caused by reconfiguration or advertisements received from other PE routers

# 3

PART

## Configuration Statements and Operational Commands

---

Configuration Statements (All VPNs) | **118**

Configuration Statements (Layer 2 VPNs) | **152**

Operational Commands | **281**

---

## Configuration Statements (All VPNs)

### IN THIS CHAPTER

- aggregate-label | 119
- backup-neighbor | 120
- description (Routing Instances) | 122
- family route-target | 123
- graceful-restart (Enabling Globally) | 125
- instance-type | 127
- interface (Routing Instances) | 130
- no-forwarding | 131
- forward-policy-mismatch (Security Group VPN Member) | 132
- proxy-generate | 133
- revert-time (Protocols Layer 2 Circuits) | 134
- route-distinguisher | 136
- route-distinguisher-id | 140
- route-target-filter | 141
- switchover-delay | 143
- unicast-reverse-path | 144
- vpn-apply-export | 145
- vrf-export | 146
- vrf-import | 148
- vrf-mtu-check | 149
- vrf-target | 150

## aggregate-label

### Syntax

```
aggregate-label {
    community community-name;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp family inet labeled-unicast],
[edit logical-systems logical-system-name protocols bgp family inet6 labeled-unicast],
[edit logical-systems logical-system-name protocols bgp family inet-vpn unicast],
[edit logical-systems logical-system-name protocols bgp family inet-vpn6 unicast],
[edit protocols bgp family inet labeled-unicast],
[edit protocols bgp family inet6 labeled-unicast],
[edit protocols bgp family inet-vpn unicast],
[edit protocols bgp family inet6-vpn unicast]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

### Description

Specify matching criteria (in the form of a community) such that all routes which match are assigned the same VPN label, selected from one of the several routes in the set defined by this criteria. This reduces the number of VPN labels that the router must consider, and aggregates the received labels.

### Options

**community *community-name***—Specify the name of the community to which to apply the aggregate label.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| [Configuring Aggregate Labels for VPNs](#)

## backup-neighbor

### Syntax

```
backup-neighbor address {
  community name;
  mtu number;
  hot-standby;
  psn-tunnel-endpoint address;
  standby;
  static;
  virtual-circuit-id number;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor address],
[edit protocols l2circuit local-switching interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name],
[edit routing-instances routing-instance-name protocols vpls neighbor address]
```

### Release Information

Statement introduced in Junos OS Release 9.2.

Statement introduced in Junos OS Release 12.2 for ACX Series routers.

Statement introduced in Junos OS Release 12.3 at the **[edit protocols l2circuit local-switching interface *interface-name*]** hierarchy level.

### Description

Configure pseudowire redundancy for Layer 2 circuits and VPLS. A redundant pseudowire can act as a backup connection and can be configured between a PE router and a CE device or between PE routers, maintaining Layer 2 circuit and VPLS services after certain types of failures. This feature can help improve the reliability of certain types of networks where a single point of failure could interrupt service for customers.

**NOTE:** VPLS is not supported on ACX Series routers. The **psn-tunnel-endpoint** statement is not supported at the **[edit protocols l2circuit local-switching interface *interface-name* end-interface *interface-name*]** hierarchy level.



**Options**

**address**—Specifies the address for the backup neighbor.

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**

---

[Configuring Pseudowire Redundancy on the PE Router | 52](#)

---

[Example: Configuring Layer 2 Circuit Switching Protection | 84](#)

---

[community \(Protocols Layer 2 Circuit\) | 159](#)

---

[psn-tunnel-endpoint | 250](#)

---

[virtual-circuit-id | 275](#)

## description (Routing Instances)

### Syntax

```
description text;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],  
[edit routing-instances routing-instance-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

Statement introduced in Junos OS Release 11.3 for the QFX Series.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Provide a text description for the routing instance. If the text includes one or more spaces, enclose it in quotation marks (" "). Any descriptive text you include is displayed in the output of the **show route instance detail** command and has no effect on the operation of the routing instance.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## family route-target

### Syntax

```
family route-target {
    advertise-default;
    external-paths number;
    prefix-limit number;
    proxy-generate <route-target-policy route-target-policy-name>;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp group group-name],
[edit logical-systems logical-system-name protocols bgp group group-name neighbor address],
[edit protocols bgp group group-name],
[edit protocols bgp group group-name neighbor address]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Enable BGP route target filtering on the VPN.

The **family route-target** statement is useful for filtering VPN routes before they are sent. Provider edge (PE) routers inform the route reflector (RR) which routes to send, using **family route-target** to provide the route-target-interest information. The RR then sends to the PE router only the advertisements containing the specified route target.

### Options

**advertise-default**—Cause the router to advertise the default route target route (**0:0:0/0**) and suppress all routes that are more specific. This can be used by a route reflector on BGP groups consisting of neighbors that act as provider edge (PE) routers only. PE routers often need to advertise all routes to the route reflector. Suppressing all route target advertisements other than the default route reduces the amount of information exchanged between the route reflector and the PE routers. The Junos OS further helps to reduce route target advertisement overhead by not maintaining dependency information unless a nondefault route is received.

**external-paths *number***—Cause the router to advertise the VPN routes that reference a given route target. The number you specify with the **external-paths** statement determines the number of external peer routers (currently advertising that route target) that receive the VPN routes. The default value is **1**.

**prefix-limit *number***—The number of prefixes that can be received from a peer router.

The remaining statement is described separately.

**Required Privilege Level**

routing—To view this statement in the configuration.

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

**RELATED DOCUMENTATION**

*Configuring BGP Route Target Filtering for VPNs*

*Understanding Proxy BGP Route Target Filtering for VPNs*

*Example: Configuring an Export Policy for BGP Route Target Filtering for VPNs*

## graceful-restart (Enabling Globally)

### Syntax

```
graceful-restart {  
  disable;  
  helper-disable;  
  maximum-helper-recovery-time seconds;  
  maximum-helper-restart-time seconds;  
  notify-duration seconds;  
  recovery-time seconds;  
  restart-duration seconds;  
  stale-routes-time seconds;  
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options],  
[edit logical-systems logical-system-name routing-instances routing-instance-name routing-options],  
[edit routing-options],  
[edit routing-instances routing-instance-name routing-options]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

You configure the graceful restart routing option globally to enable the feature, but not to enable graceful restart for all routing protocols in a routing instance. To enable graceful restart globally, include the graceful-restart statement under the **[edit routing options]** hierarchy level. This enables graceful restart globally for all routing protocols. You can, optionally, modify the global settings at the individual protocol level.

**NOTE:**

- For VPNs, the **graceful-restart** statement allows a router whose VPN control plane is undergoing a restart to continue to forward traffic while recovering its state from neighboring routers.
- For BGP, if you configure graceful restart after a BGP session has been established, the BGP session restarts and the peers negotiate graceful restart capabilities.
- LDP sessions flap when **graceful-restart** configurations change.

**Default**

Graceful restart is disabled by default.

**Options**

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**

---

*Enabling Graceful Restart*

---

*Configuring Routing Protocols Graceful Restart*

---

*Configuring Graceful Restart for MPLS-Related Protocols*

---

*Configuring VPN Graceful Restart*

---

*Configuring Logical System Graceful Restart*

---

*Configuring Graceful Restart for QFabric Systems*

## instance-type

### Syntax

```
instance-type type;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],  
[edit routing-instances routing-instance-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

**virtual-switch** and **layer2-control** options introduced in Junos OS Release 8.4.

Statement introduced in Junos OS Release 9.2 for EX Series switches.

Statement introduced in Junos OS Release 11.3 for the QFX Series.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

**mpls-internet-multicast** option introduced in Junos OS Release 11.1 for the EX Series, M Series, MX Series, and T Series.

**evpn** option introduced in Junos OS Release 13.2 for MX 3D Series routers.

**evpn** option introduced in Junos OS Release 17.3 for the QFX Series.

**forwarding** option introduced in Junos OS Release 14.2 for the PTX Series.

**mpls-forwarding** option introduced in Junos OS Release 16.1 for the MX Series.

**evpn-vpws** option introduced in Junos OS Release 17.1 for MX Series routers.

Support for logical systems on MX Series routers added in Junos OS Release 17.4R1.

### Description

Define the type of routing instance.

### Options

**NOTE:** On ACX Series routers, you can configure only the forwarding, virtual router, and VRF routing instances.

**type**—Can be one of the following:

- **evpn**—(MX 3D Series routers, QFX switches and EX9200 switches)—Enable an Ethernet VPN (EVPN) on the routing instance.  
hierarchy level.
- **evpn-vpws**—Enable an Ethernet VPN (EVPN) Virtual Private Wire Service (VPWS) on the routing instance.

- **forwarding**—Provide support for filter-based forwarding, where interfaces are not associated with instances. All interfaces belong to the default instance. Other instances are used for populating RPD learned routes. For this instance type, there is no one-to-one mapping between an interface and a routing instance. All interfaces belong to the default instance inet.0.
- **l2backhaul-vpn**—Provide support for Layer 2 wholesale VLAN packets with no existing corresponding logical interface. When using this instance, the router learns both the outer tag and inner tag of the incoming packets, when the **instance-role** statement is defined as **access**, or the outer VLAN tag only, when the **instance-role** statement is defined as **nni**.
- **l2vpn**—Enable a Layer 2 VPN on the routing instance. You must configure the **interface**, **route-distinguisher**, **vrf-import**, and **vrf-export** statements for this type of routing instance.
- **layer2-control**—(MX Series routers only) Provide support for RSTP or MSTP in customer edge interfaces of a VPLS routing instance. This instance type cannot be used if the customer edge interface is multihomed to two provider edge interfaces. If the customer edge interface is multihomed to two provider edge interfaces, use the default BPDU tunneling.
- **mpls-forwarding**—(MX Series routers only) Allow filtering and translation of route distinguisher (RD) values in IPv4 and IPv6 VPN address families on both routes received and routes sent for selected BGP sessions. In particular, for Inter-AS VPN Option-B networks, this option can prevent the malicious injection of VPN labels from one peer AS boundary router to another.
- **mpls-internet-multicast**—(EX Series, M Series, MX Series, and T Series routers only) Provide support for ingress replication provider tunnels to carry IP multicast data between routers through an MPLS cloud, using MBGP or next-generation MVPN.
- **no-forwarding**—This is the default routing instance. Do not create a corresponding forwarding instance. Use this routing instance type when a separation of routing table information is required. There is no corresponding forwarding table. All routes are installed into the default forwarding table. IS-IS instances are strictly nonforwarding instance types.
- **virtual-router**—Enable a virtual router routing instance. This instance type is similar to a VPN routing and forwarding instance type, but used for non-VPN-related applications. You must configure the **interface** statement for this type of routing instance. You do not need to configure the **route-distinguisher**, **vrf-import**, and **vrf-export** statements.
- **virtual-switch**—(MX Series routers, EX9200 switches, and QFX switches only) Provide support for Layer 2 bridging. Use this routing instance type to isolate a LAN segment with its Spanning Tree Protocol (STP) instance and to separate its VLAN identifier space.



- **vpls**—Enable VPLS on the routing instance. Use this routing instance type for point-to-multipoint LAN implementations between a set of sites in a VPN. You must configure the **interface**, **route-distinguisher**, **vrf-import**, and **vrf-export** statements for this type of routing instance.
- **vrf**—VPN routing and forwarding (VRF) instance. Provides support for Layer 3 VPNs, where interface routes for each instance go into the corresponding forwarding table only. Required to create a Layer 3 VPN. Create a VRF table (**instance-name.inet.0**) that contains the routes originating from and destined for a particular Layer 3 VPN. For this instance type, there is a one-to-one mapping between an interface and a routing instance. Each VRF instance corresponds with a forwarding table. Routes on an interface go into the corresponding forwarding table. You must configure the **interface**, **route-distinguisher**, **vrf-import**, and **vrf-export** statements for this type of routing instance.

#### Required Privilege Level

routing—To view this statement in the configuration.

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

#### RELATED DOCUMENTATION

*Configuring the Instance Type*

*Configuring EVPN Routing Instances*

*Configuring EVPN Routing Instances on EX9200 Switches*

*Configuring Virtual Router Routing Instances*

*Example: Configuring Filter-Based Forwarding on the Source Address*

*Example: Configuring Filter-Based Forwarding on Logical Systems*

## interface (Routing Instances)

### Syntax

```
interface interface-name {
    description text;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],
[edit routing-instances routing-instance-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.2 for EX Series switches.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Statement introduced in Junos OS Release 13.2 for MX 3D Series routers.

### Description

Specify the interface over which the VPN traffic travels between the PE device and CE device. You configure the interface on the PE device. If the value **vrf** is specified for the **instance-type** statement included in the routing instance configuration, this statement is required.

### Options

***interface-name***—Name of the interface.

The remaining statement is explained separately. Search for a statement in [CLI Explorer](#) or click a linked statement in the Syntax section for details.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring Interfaces for VPN Routing](#)

[Configuring EVPN Routing Instances](#)

[Configuring EVPN Routing Instances on EX9200 Switches](#)

[interface \(VPLS Routing Instances\)](#)

## no-forwarding

### Syntax

```
no-forwarding;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols ldp],  
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols ldp],  
[edit protocols ldp],  
[edit routing-instances routing-instance-name protocols ldp]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 12.3X50 for the QFX Series.

### Description

Do not add ingress routes to the inet.0 routing table even if **traffic-engineering bgp-igp** (configured at the **[edit protocols mpls]** hierarchy level) is enabled.

### Default

The **no-forwarding** statement is disabled. Ingress routes are added to the inet.0 routing table instead of the inet.3 routing table when **traffic-engineering bgp-igp** is enabled.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Preventing Addition of Ingress Routes to the inet.0 Routing Table*

*Configuring Virtual-Router Routing Instances in VPNs*

## forward-policy-mismatch (Security Group VPN Member)

### Syntax

```
vpn vpn-name {  
    ike-gateway gateway-number;  
    group group-number;  
    match-direction (input);  
    tunnel mtu mtu-size;  
}
```

### Hierarchy Level

```
[edit security group-vpn member ipsec]
```

### Release Information

Statement introduced in Junos OS Release 18.2R1.

### Description

Configure the forward policy mismatch for group VPN member.

### Required Privilege Level

security—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| [Group VPNv2 Overview](#)

## proxy-generate

### Syntax

```
proxy-generate <route-target-policy route-target-policy-name>;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp group group-name family route-target],
[edit logical-systems logical-system-name protocols bgp group group-name neighbor address family route-target],
[edit protocols bgp group group-name family route-target],
[edit protocols bgp group group-name neighbor address family route-target]
```

### Release Information

Statement introduced in Junos OS Release 12.2.

### Description

Enable proxy BGP route target filtering (also known as proxy route target constrain, or proxy RTC). This feature is useful if you have a network environment where route target filtering is not widely deployed or fully supported. When configured for proxy BGP route target filtering, the device creates route target membership (RT membership) on behalf of its peers that do not have the route target filtering functionality. The device then distributes the RT membership advertisements from incoming BGP VPN routes to other devices in the network that need them.

### Options

**route-target-policy *route-target-policy-name***—(Optional) Apply a routing policy that defines a subset of VPN routes to be used in your proxy BGP route target filter. The subset of VPN routes control which proxy BGP route targets are used to create the proxy BGP route target routes.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Example: Configuring Proxy BGP Route Target Filtering for VPNs](#)

[Example: Configuring an Export Policy for BGP Route Target Filtering for VPNs](#)

[Configuring BGP Route Target Filtering for VPNs](#)

[family route-target](#) | 123

[rtf-prefix-list](#)

## revert-time (Protocols Layer 2 Circuits)

### Syntax

```
revert-time seconds maximum seconds;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],  
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor address],  
[edit protocols l2circuit neighbor address interface interface-name],  
[edit routing-instances routing-instance-name protocols vpls neighbor address]
```

### Release Information

Statement introduced in Junos OS Release 10.2.

**maximum** option introduced in Junos OS Release 13.2.

### Description

Specify a revert time for redundant Layer 2 circuits and VPLS pseudowires. When you have configured redundant pseudowires for Layer 2 circuits or VPLS, traffic is switched to the backup connection in the event that the primary connection fails. If you configure a revert time, when the configured time expires traffic is reverted to the primary path, assuming the primary path has been restored.

With the **maximum** option, specify a maximum reversion interval to add after the **revert-time** delay. If a revert-time delay is defined but a maximum timer is not defined, VCs are restored upon the revert-timer's expiration.

To reduce as much as possible the amount of traffic discarded, and potential data-path asymmetries observed during primary-to-backup transition periods, you can use this restoration timer. This restoration timer is activated when the backup path is performing as active, and then the primary path is restored. The goal is to avoid moving traffic back to the primary path right away, to make sure that the control plane's related tasks (such as IGP, LDP, RSVP, and internal BGP) have enough time to complete their updating cycle.

By enabling a gradual return of traffic to the primary path, you can ensure that the relatively-slow control-plane processing and updating does not have a negative impact on the restoration process.

The **maximum** option extends the revert timer's functionality to provide a jittered interval over which a certain number of circuits can be transitioned back to the primary path. By making use of this maximum value, you can define a time interval during which circuits are expected to switch over. As a consequence, circuits' effective transitions are scattered during restoration periods.

When making use of **revert-time x maximum y** statement, you can ensure that the corresponding circuit that is active is moved to the primary path within a time-slot ( $t_1$ ) such as that:  $x \leq t_1 \leq y$ . In other words, by activating this statement, you can ensure the following:

- VCs stay in the backup path for at least  $x$  seconds after the primary path comes back up.
- VCs are moved back to the primary path before  $y$  seconds have elapsed.
- $y$  maximum value =  $x$  maximum value \* 2 = 1200 seconds.

The ideal values for  $x$  and  $y$  will be conditioned to internal aspects of your network. For this reason, there are no default values for these settings. If no revert-time is set, the default behavior is non-revertive. That is, circuits are not returned to the primary path upon restoration. They are kept on the backup path.

### Default

Without the **revert-time** statement, virtual circuit (VC) traffic is not transitioned to the primary path upon restoration of the primary path.

### Options

*seconds*—Revert time in seconds.

**Range:** 0 through 600 seconds

maximum *seconds*—Number of seconds to delay path restoration after the **revert-time** delay.

**Range:** 0 through 1200 seconds

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| [Configuring Redundant Pseudowires for Layer 2 Circuits and VPLS](#) | 52

## route-distinguisher

### Syntax

```
route-distinguisher (as-number:id | ip-address:id);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn mesh-group
  mesh-group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls mesh-group
  mesh-group-name],
[edit routing-instances routing-instance-name],
[edit routing-instances routing-instance-name protocols l2vpn mesh-group mesh-group-name],
[edit routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

Support at **[edit routing-instances *routing-instance-name* protocols vpls mesh-group *mesh-group-name*]**

hierarchy level introduced in Junos OS Release 11.2.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Support at **[edit routing-instances *routing-instance-name* protocols l2vpn mesh-group *mesh-group-name*]**

hierarchy level introduced in Junos OS Release 13.2.

Statement introduced in Junos OS Release 14.1X53-D30 for QFX Series switches.

Statement introduced in cRPD Release 19.4R1.

### Description

Specify an identifier attached to a route, enabling you to distinguish to which VPN or virtual private LAN service (VPLS) the route belongs. Each routing instance must have a unique route distinguisher (RD) associated with it. The RD is used to place bounds around a VPN so that the same IP address prefixes can be used in different VPNs without having them overlap. If the instance type is **vrf**, the **route-distinguisher** statement is required.

For Layer 2 VPNs and VPLS, if you configure the **l2vpn-use-bgp-rules** statement, you must configure a unique RD for each PE router participating in the routing instance.

For other types of VPNs, we recommend that you use a unique RD for each provider edge (PE) router participating in specific routing instance. Although you can use the same RD on all PE routers for the same VPN routing instance, if you use a unique RD, you can determine the customer edge (CE) router from which a route originated within the VPN.

For Layer 2 VPNs and VPLSs, if you configure mesh groups, the RD in each mesh group must be unique.





**CAUTION:** We strongly recommend that if you change an RD that has already been configured, or change the routing-instance type from **virtual-router** to **vrf**, make the change during a maintenance window, as follows:

1. Deactivate the routing instance.
2. Change the RD.
3. Activate the routing instance.

This is not required if you are configuring the RD for the first time.

## Options

**as-number:number—as-number** is an assigned AS number, and **number** is any 2-byte or 4-byte value. The AS number can be from 1 through 4,294,967,295. If the AS number is a 2-byte value, the administrative number is a 4-byte value. If the AS number is 4-byte value, the administrative number is a 2-byte value. An RD consisting of a 4-byte AS number and a 2-byte administrative number is defined as a type 2 RD in RFC 4364 *BGP/MPLS IP VPNs*.

**NOTE:** In Junos OS Release 9.1 and later, the numeric range for AS numbers is extended to provide BGP support for 4-byte AS numbers, as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*. All releases of Junos OS support 2-byte AS numbers. To configure an RD that includes a 4-byte AS number, append the letter “L” to the end of the AS number. For example, an RD with the 4-byte AS number 7,765,000 and an administrative number of 1,000 is represented as **77765000L:1000**.

In Junos OS Release 9.2 and later, you can also configure a 4-byte AS number using the AS dot notation format of two integer values joined by a period: *<16-bit high-order value in decimal>.<16-bit low-order value in decimal>*. For example, the 4-byte AS number of 65,546 in the plain-number format is represented as 1.10 in AS dot notation format.

**ip-address:id**—IP address (**ip-address** is a 4-byte value) within your assigned prefix range and a 2-byte value for the **id**. The IP address can be any globally unique unicast address.

**Range:** 0 through 4,294,967,295 ( $2^{32} - 1$ ). If the router you are configuring is a BGP peer of a router that does not support 4-byte AS numbers, you need to configure a local AS number. For more information, see *Using 4-Byte Autonomous System Numbers in BGP Networks Technology Overview*.

**NOTE:** For Ethernet VPN (EVPN), an RD that includes zero as the **id** value is reserved for the default EVPN routing instance by default. Because the same RD cannot be assigned for two routing instances, using a **ip-address:id** RD for another routing instance (default-switch), where the **id** value is zero, throws a commit error.

## Required Privilege Level

routing—To view this statement in the configuration.

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

RELATED DOCUMENTATION

<i>Example: Configuring BGP Route Target Filtering for VPNs</i>
<i>Example: Configuring FEC 129 BGP Autodiscovery for VPWS</i>
<i>Configuring EVPN Routing Instances</i>
<i>Configuring Routing Instances on PE Routers in VPNs</i>
<i>Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)</i>
<i>Configuring an MPLS-Based Layer 3 VPN (CLI Procedure)</i>
<a href="#">path-selection</a>   <b>240</b>

## route-distinguisher-id

### Syntax

```
route-distinguisher-id ip-address;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options],  
[edit routing-options]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

### Description

Automatically assign a route distinguisher to the routing instance.

If you configure the **route-distinguisher** statement in addition to the **route-distinguisher-id** statement, the value configured for **route-distinguisher** supersedes the value generated from **route-distinguisher-id**.

**NOTE:** To avoid a conflict in the two route distinguisher values, it is recommended to ensure that the first half of the route distinguisher obtained by configuring the **route-distinguisher** statement is different from the first half of the route distinguisher obtained by configuring the **route-distinguisher-id** statement.

### Options

***ip-address***—Address for routing instance.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Example: Configuring BGP Route Target Filtering for VPNs*

*Configuring Routing Instances on PE Routers in VPNs*

## route-target-filter

### Syntax

```
route-target-filter destination {
    group group-name;
    local;
    neighbor address;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options rib bgp.rtarget.0 static],
[edit routing-options rib bgp.rtarget.0 static]
```

### Release Information

Statement introduced in Junos OS Release 12.2.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

### Description

Statically configure route target filtering. Route target filtering allows you to distribute VPN routes to only the routers that need them. In VPN networks without route target filtering configured, BGP distributes all static VPN routes to all of the VPN peer routers. You can add static routes to the bgp.rtarget.0 routing table with specific NLRI-imposed constraints.

### Options

**destination**—Allows you to specify the static route destination. This value must be in the format x:y/len.

The x value is either an IP address or an AS number followed by an optional L to indicate a 4 byte AS number, and y is a number (for example, 123456L:100/64).

**group group-name(s)**—Installs an RT-Constrain filter for the destination for all peers in the specified BGP group. The route and corresponding BGP group are displayed in the output of the **show bgp group rtf detail** command.

**local**—Causes the router to originate the route target constrain NLRI, but does not install any filtering state for the prefix. This behavior can be useful when the router should always receive VPN routes with this route-target regardless of the state of a given BGP peering session or group status. The route is not displayed in the output of the **show bgp group rtf detail** command unless it is also included in either the BGP neighbor or BGP group configuration.

**neighbor address**—Installs an RT-Constrain filter for the destination for this BGP neighbor. The route is displayed in the output of the **show bgp group rtf detail** command. Specify the BGP neighbor using the router's IP address.

**Required Privilege Level**

routing—To view this statement in the configuration.

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

**RELATED DOCUMENTATION**

*Configuring Static Route Target Filtering for VPNs*

*Reducing Network Resource Use with Static Route Target Filtering for VPNs*

## switchover-delay

### Syntax

```
switchover-delay milliseconds;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],  
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor address],  
[edit protocols l2circuit neighbor address interface interface-name],  
[edit routing-instances routing-instance-name protocols vpls neighbor address]
```

### Release Information

Statement introduced in Junos OS Release 9.2.

### Description

After the primary pseudowire goes down, specifies the delay (in milliseconds) to wait before the backup pseudowire takes over. You configure this statement for each backup neighbor configuration to adjust the switchover time after a failure is detected.

### Options

***milliseconds***—Specify the time to wait before switching to the backup pseudowire after the primary pseudowire fails.

**Default:** 10,000 milliseconds

**Range:** 0 through 180,000 milliseconds

### Required Privilege Level

routing—To view this statement in the configuration.

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

### RELATED DOCUMENTATION

[Configuring the Switchover Delay for the Pseudowires](#) | 53

## unicast-reverse-path

### Syntax

```
unicast-reverse-path (active-paths | feasible-paths);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options forwarding-table],  
[edit routing-instances routing-instance-name instance-type name routing-options forwarding-table],  
[edit routing-options forwarding-table]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Support for routing instances added in Junos OS Release 8.3.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Statement introduced in Junos OS Release 11.3 for QFX Series switches.

**NOTE:** This feature is not supported on the EX4300 switch, even though it is available on the device.

### Description

Control the operation of unicast reverse-path-forwarding check. This statement enables the RPF check to be used when routing is asymmetrical.

### Options

**active-paths**—Consider only active paths during the unicast reverse-path check.

**feasible-paths**—Consider all feasible paths during the unicast reverse-path check.

**Default:** If you omit the **unicast-reverse-path** statement, only the active paths to a particular destination are considered.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Example: Configuring Unicast RPF (On a Router)*



## vpn-apply-export

### Syntax

```
vpn-apply-export;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp],  
[edit logical-systems logical-system-name protocols bgp group group-name],  
[edit logical-systems logical-system-name protocols bgp group group-name neighbor neighbor],  
[edit protocols bgp],  
[edit protocols bgp group group-name],  
[edit protocols bgp group group-name neighbor neighbor]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Apply both the VRF export and BGP group or neighbor export policies (VRF first, then BGP) before routes from the **vrf** or **I2vpn** routing tables are advertised to other PE routers.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

Configuring Policies for the VRF Table on PE Routers in VPNs

## vrf-export

### Syntax

```
vrf-export [ policy-names ];
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls mesh-group
  mesh-group-name]
[edit routing-instances routing-instance-name]
[edit routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name]
[edit switch-options]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Statement introduced in Junos OS Release 14.1X53-D30 for QFX Series switches.

Statement introduced in cRPD Release 19.4R1.

### Description

Specify how routes are exported from the local PE router's VRF table (*routing-instance-name*.inet.0) to the remote PE router. If the value **vrf** is specified for the **instance-type** statement included in the routing instance configuration, this statement is required.

You can configure multiple export policies on the PE router or PE switch.

### Default

If the instance-type is **vrf**, **vrf-export** is a required statement. The default action is to reject.

### Options

***policy-names***— Names for the export policies.

### Required Privilege Level

routing— To view this statement in the configuration.

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

## RELATED DOCUMENTATION

Implementing EVPN-VXLAN for Data Centers

[instance-type](#) | 127

---

*Configuring Policies for the VRF Table on PE Routers in VPNs*

## vrf-import

### Syntax

```
vrf-import [ policy-names ];
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls mesh-group
  mesh-group-name]
[edit routing-instances routing-instance-name]
[edit routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name]
[edit switch-options]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

Statement introduced in Junos OS Release 14.1X53-D30 for QFX Series switches.

Statement introduced in cRPD Release 19.4R1.

### Description

Specify how routes are imported into the virtual routing and forwarding (VRF) table (***routing-instance-name***.inet.0) of the local provider edge (PE) router or switch from the remote PE router. If the value **vrf** is specified for the **instance-type** statement included in the routing instance configuration, this statement is required.

You can configure multiple import policies on the PE router or switch.

### Default

If the instance type is **vrf**, **vrf-import** is a required statement. The default action is to accept.

### Options

***policy-names***—Names for the import policies.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

Implementing EVPN-VXLAN for Data Centers

## vrf-mtu-check

### Syntax

```
vrf-mtu-check;
```

### Hierarchy Level

```
[edit chassis]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

### Description

On M Series routers (except the M120 and M320 router) and on EX Series 8200 switches, configure path maximum transmission unit (MTU) checks on the outgoing interface for unicast traffic routed on a virtual private network (VPN) routing and forwarding (VRF) instance.

### Default

Disabled.

### Required Privilege Level

interface—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Configuring Path MTU Checks for VPN Routing Instances*

*Configuring the Junos OS to Enable MTU Path Check for a Routing Instance on M Series Routers*

## vrf-target

### Syntax

```
vrf-target {
  community;
  auto
  import community-name;
  export community-name;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn mesh-group
  mesh-group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls mesh-group
  mesh-group-name],
[edit routing-instances routing-instance-name protocols evpn vni-options],
[edit routing-instances routing-instance-name protocols l2vpn mesh-group mesh-group-name],
[edit routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name],
[edit switch-options]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Statement introduced in Junos OS Release 14.1X53-D30 for QFX Series switches. **auto option** was also added at this time.

**auto option** added in Junos OS Release 19.1R1 for MX series.

Statement introduced in cRPD Release 19.4R1.

### Description

Specify a virtual routing and forwarding (VRF) target community. If you configure the **community** option only, default VRF import and export policies are generated that accept and tag routes with the specified target community. The purpose of the **vrf-target** statement is to simplify the configuration by allowing you to configure most statements at the **[edit routing-instances]** hierarchy level. In effect, this statement configures a single policy for import and a single policy for export to replace the per-VRF policies for every community.

You can still create more complex policies by explicitly configuring VRF import and export policies using the **import** and **export** options.

## Options

**community**—Community name.

**auto**—Automatically derives the route target (RT). The auto-derived route targets have higher precedence over manually configured RT in vrf-target, vrf-export policies, and vrf-import policies.

**NOTE:** Auto-derived route targets are supported only in virtual switch and EVPN routing instances.

**import community-name**—Allowed communities accepted from neighbors.

**export community-name**—Allowed communities sent to neighbors.

## Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Configuring Policies for the VRF Table on PE Routers in VPNs*

*Example: Configuring FEC 129 BGP Autodiscovery for VPWS*

## Configuration Statements (Layer 2 VPNs)

### IN THIS CHAPTER

- auto-discovery-only | 155
- backup-interface (Layer 2 Circuits) | 156
- bfd-liveness-detection (Layer 2 VPN and VPLS) | 157
- community (Protocols Layer 2 Circuit) | 159
- connection-protection | 160
- control-channel (Protocols OAM) | 161
- control-word (Protocols Layer 2 Circuit Neighbor) | 162
- control-word (Protocols Layer 2 VPN) | 163
- description (Protocols Layer 2 Circuit Neighbor) | 164
- description (Protocols Layer 2 VPN) | 165
- detection-time (BFD Liveness Detection) | 166
- egress-protection (Layer 2 circuit) | 169
- egress-protection (MPLS) | 170
- encapsulation (Logical Interface) | 172
- encapsulation | 177
- encapsulation-type (Layer 2 Circuits) | 184
- encapsulation-type (Layer 2 VPNs) | 186
- end-interface | 188
- family (Protocols BGP) | 189
- family multiservice | 195
- flow-label-receive-static | 198
- flow-label-transmit-static | 199
- hot-standby | 200
- hot-standby (Protocols Layer 2 Circuit) | 201
- hot-standby-vc-on (Protocols Layer 2 Circuit) | 202
- ignore-encapsulation-mismatch | 204
- ignore-mtu-mismatch | 205
- interface (Protocols Layer 2 Circuit) | 206



- interface (Protocols Layer 2 VPN) | 208
- install-nexthop | 209
- l2circuit | 210
- l2ckt | 212
- l2vpn | 213
- l2vpn (routing-options) | 216
- l2vpn-id | 217
- local-switching (Layer 2 Circuits) | 218
- minimum-interval (BFD Liveness Detection) | 219
- minimum-receive-interval (BFD Liveness Detection) | 221
- mtu | 223
- multiplier (BFD Liveness Detection) | 227
- neighbor (Protocols Layer 2 Circuit) | 229
- no-adaptation (BFD Liveness Detection) | 231
- no-control-word (Protocols Layer 2 VPN) | 233
- no-l2ckt | 234
- no-l2vpn | 235
- no-revert (Local Switching) | 236
- no-revert (Neighbor Interface) | 237
- oam | 238
- path-selection | 240
- ping-interval | 243
- policer (Layer 2 VPN) | 244
- protect-interface | 245
- protected-l2circuit | 246
- protector-interface | 247
- protector-pe | 248
- pseudowire-status-tlv | 249
- psn-tunnel-endpoint | 250
- remote-site-id | 251
- routing-instances | 252
- send-oam | 253
- site (Layer 2 Circuits) | 254
- site-identifier (Layer 2 Circuits) | 255

- [site-preference](#) | **256**
- [source-attachment-identifier \(Protocols VPWS\)](#) | **257**
- [standby \(Protocols Layer 2 Circuit\)](#) | **259**
- [static \(Protocols Layer 2 Circuit\)](#) | **260**
- [target-attachment-identifier \(Protocols VPWS\)](#) | **262**
- [template](#) | **263**
- [traceoptions \(Egress Protection\)](#) | **264**
- [traceoptions \(Protocols Layer 2 Circuit\)](#) | **266**
- [traceoptions \(Protocols Layer 2 VPN\)](#) | **268**
- [transmit-interval \(BFD Liveness Detection\)](#) | **270**
- [version \(BFD Liveness Detection\)](#) | **273**
- [virtual-circuit-id](#) | **275**
- [vlan-id](#) | **276**
- [vlan-id \(routing instance\)](#) | **277**
- [vlan-tagging](#) | **278**

## auto-discovery-only

### Syntax

```
auto-discovery-only;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp family l2vpn],
[edit logical-systems logical-system-name protocols bgp group group-name family l2vpn],
[edit logical-systems logical-system-name protocols bgp group group-name neighbor address family l2vpn],
[edit logical-systems logical-system-name routing-instances instance-name protocols bgp family l2vpn],
[edit logical-systems logical-system-name routing-instances instance-name protocols bgp group group-name family
  l2vpn],
[edit logical-systems logical-system-name routing-instances instance-name protocols bgp group group-name neighbor
  address family l2vpn],
[edit protocols bgp family l2vpn],
[edit protocols bgp group group-name family l2vpn],
[edit protocols bgp group group-name neighbor address family l2vpn],
[edit routing-instances instance-name protocols bgp family l2vpn],
[edit routing-instances instance-name protocols bgp group group-name family l2vpn],
[edit routing-instances instance-name protocols bgp group group-name neighbor address family l2vpn]
```

### Release Information

Statement introduced in Junos OS Release 10.4R2.

### Description

Enable the router to process only the autodiscovery network layer reachability information (NLRI) update messages for VPWS and LDP-based Layer 2 VPN and VPLS update messages (BGP\_L2VPN\_AD\_NLRI) (FEC 129).

Specifically, the **auto-discovery-only** statement notifies the routing process (rpd) to expect autodiscovery-related NLRI messages so that information can be deciphered and used by LDP, VPLS, and VPWS.

The **auto-discovery-only** statement must be configured on all provider edge (PE) routers in a VPLS or in a VPWS. If you configure route reflection, the **auto-discovery-only** statement is also required on provider (P) routers that act as the route reflector in supporting FEC 129-related updates.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Example: Configuring BGP Autodiscovery for LDP VPLS*

*Example: Configuring BGP Autodiscovery for LDP VPLS with User-Defined Mesh Groups*

*Example: Configuring FEC 129 BGP Autodiscovery for VPWS*

## backup-interface (Layer 2 Circuits)

### Syntax

```
backup-interface interface-name;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name end-interface
  interface interface-name],
[edit protocols l2circuit local-switching interface interface-name end-interface interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 12.3.

### Description

Specify the interface to be used by the protection pseduowire in connection protection configurations for Layer 2 circuits.

### Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration

## RELATED DOCUMENTATION

[Example: Configuring Layer 2 Circuit Switching Protection](#) | 84

## bfd-liveness-detection (Layer 2 VPN and VPLS)

### Syntax

```

bfd-liveness-detection {
  detection-time {
    threshold milliseconds;
  }
  minimum-interval milliseconds;
  minimum-receive-interval milliseconds;
  multiplier number;
  no-adaptation;
  transmit-interval {
    minimum-interval milliseconds;
    threshold milliseconds;
  }
  version (1 | automatic);
}

```

### Hierarchy Level

```

[edit logical-system logical-system-name routing-instances routing-instance-name protocols l2vpn oam],
[edit logical-system logical-system-name routing-instances routing-instance-name protocols vpls neighbor neighbor-id oam],
[edit logical-system logical-system-name routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name neighbor neighbor-id oam],
[edit logical-system logical-system-name routing-instances routing-instance-name protocols vpls oam],
[edit routing-instances routing-instance-name protocols l2vpn oam],
[edit routing-instances routing-instance-name protocols vpls neighbor neighbor-id oam],
[edit routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name neighbor neighbor-id oam],
[edit routing-instances routing-instance-name protocols vpls oam]

```

### Release Information

Statement introduced in Junos OS Release 10.0.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPNs and VPLS.

### Description

Configure bidirectional failure detection timers.

The BFD failure detection timers are adaptive and can be adjusted to be more or less aggressive. For example, the timers can adapt to a higher value if the adjacency fails, or a neighbor can negotiate a higher value for a timer than the configured value. The timers adapt to a higher value when a BFD session flap occurs more than three times in a span of 15 seconds. A back-off algorithm increases the receive (Rx)

interval by two if the local BFD instance is the reason for the session flap. The transmission (Tx) interval is increased by two if the remote BFD instance is the reason for the session flap. You can use the **clear bfd adaptation** command to return BFD interval timers to their configured values. The **clear bfd adaptation** command is hitless, meaning that the command does not affect traffic flow on the routing device.

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

[Configuring BFD for Layer 2 VPN and VPLS | 56](#)

*Example: Configuring BFD for Static Routes for Faster Network Failure Detection*

## community (Protocols Layer 2 Circuit)

### Syntax

```
community community-name {
    invert-match;
    members community-members;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name policy-options],
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name backup-neighbor
address],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name backup-neighbor
address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor address
backup-neighbor address],
[edit policy-options],
[edit protocols l2circuit local-switching interface interface-name backup-neighbor address],
[edit protocols l2circuit neighbor address interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name backup-neighbor address],
[edit routing-instances routing-instance-name protocols vpls neighbor address backup-neighbor address]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Hierarchy levels associated with the **backup-neighbor** statement (pseudowire redundancy) added in Junos OS Release 9.2.

Statement introduced in Junos OS Release 12.3 at the **[edit protocols l2circuit local-switching interface *interface-name* backup-neighbor *address*]** hierarchy level.

### Description

Specify the community for the Layer 2 circuit.

### Options

***community-name***—Name of the Layer 2 circuit community.

***invert-match***—Invert the results of the community expression match.

***members community-members***—Specify the members of the community.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring the Layer 2 Circuit Community | 78](#)

[Configuring Pseudowire Redundancy on the PE Router | 52](#)

[Example: Configuring Layer 2 Circuit Switching Protection | 84](#)

## connection-protection

### Syntax

```
connection-protection;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface],
[edit protocols l2circuit local-switching interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 12.3.

### Description

Enable connection protection on the Layer 2 circuit. Connection protection enables you to configure a redundant pseudowire to act as a backup connection and can be configured between PE routers, maintaining Layer 2 circuit and VPLS services after certain types of failures. This feature helps to improve the reliability of networks where a single point of failure could interrupt service for customers.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Example: Configuring Layer 2 Circuit Switching Protection | 84](#)



## control-channel (Protocols OAM)

### Syntax

```
control-channel {
  pwe3-control-word;
  pw-label-ttl-1;
  router-alert-label;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn oam],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls oam],
[edit routing-instances routing-instance-name protocols l2vpn oam],
[edit routing-instances routing-instance-name protocols vpls oam]
```

### Release Information

Statement introduced in Junos OS Release 10.0.

### Description

Configure the Virtual Circuit Connection Verification (VCCV) BFD control channel. VCCV provides a control channel associated with a pseudowire. You can configure a number of different CV types for this control channel, based on the configuration of the pseudowire.

### Options

**pwe3-control-word**—For BGP-based pseudowires that send OAM packets with a control word that has 0001b as the first nibble.

**pw-label-ttl-1**—For BGP-based pseudowires that send OAM packets with the MPLS pseudowire label and time-to-live (TTL) set to 1.

**router-alert-label**—For BGP-based pseudowires that send OAM packets with router alert label.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring BFD for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS](#) | 59

## control-word (Protocols Layer 2 Circuit Neighbor)

### Syntax

```
(control-word | no-control-word);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],  
[edit protocols l2circuit neighbor address interface interface-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Specify the control word. The control word is four bytes long and is inserted between the Layer 2 protocol data unit (PDU) being transported and the virtual circuit (VC) label that is used for demultiplexing.

### Options

**control-word**—Enable the use of the control word.

**Default:** A null control word is enabled by default. You can also configure the control word explicitly using the **control-word** statement.

**no-control-word**—Disable the use of the control word.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring the Control Word for Frame Relay Interfaces](#) | 71

## control-word (Protocols Layer 2 VPN)

### Syntax

```
control-word;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn],  
[edit routing-instances routing-instance-name protocols l2vpn]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Specify the control word. The control word is 4 bytes long and is inserted between the Layer 2 protocol data unit (PDU) being transported and the virtual connection (VC) label that is used for demultiplexing.

**NOTE:** The following configuration statements are ignored for time-division multiplexing pseudowires at the **[edit protocols l2vpn]** hierarchy level:

- control-word
- no-control-word

### Default

The control word is enabled by default. You can also configure the control word explicitly using the **control-word** statement.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Disabling the Control Word for Layer 2 VPNs | 38](#)

[no-control-word | 233](#)

## description (Protocols Layer 2 Circuit Neighbor)

### Syntax

```
description text;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],  
[edit protocols l2circuit neighbor address interface interface-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Provide a text description for the Layer 2 circuit. If the text includes one or more spaces, enclose the entire text string in quotation marks (" ").

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| *Configuring Routing Instances on PE Routers in VPNs*

## description (Protocols Layer 2 VPN)

### Syntax

```
description text;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn site site-name
  interface interface-name],
[edit routing-instances routing-instance-name protocols l2vpn site site-name interface interface-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

### Description

Describe the VPN or virtual private LAN service (VPLS) routing instance.

### Options

**text**—Provide a text description. If the text includes one or more spaces, enclose it in quotation marks (" "). Any descriptive text you include is displayed in the output of the **show route instance detail** command and has no effect on operation.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring the Site | 11](#)

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## detection-time (BFD Liveness Detection)

### Syntax

```
detection-time {
    threshold milliseconds;
}
```

### BGP

```
[edit logical-systems name protocols bgp bfd-liveness-detection],
[edit logical-systems name protocols bgpgroup bfd-liveness-detection],
[edit logical-systems name protocols bgp group name neighbor address bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols bgp bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols bgpgroup bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols bgpgroup neighbor address bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols bgp bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols bgpgroup bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols bgpgroup neighbor address
    bfd-liveness-detection],
[edit protocols bgp bfd-liveness-detection],
[edit protocols bgp group bfd-liveness-detection],
[edit protocols bgp group neighbor address bfd-liveness-detection],
[edit routing-instances name protocols bgp bfd-liveness-detection],
[edit routing-instances name protocols bgp group bfd-liveness-detection],
[edit routing-instances name protocols bgp group neighbor address bfd-liveness-detection],
[edit tenants name routing-instances name protocols bgp bfd-liveness-detection]
[edit tenants name routing-instances name protocols bgp group bfd-liveness-detection]
[edit tenants name routing-instances name protocols bgp groupneighbor address bfd-liveness-detection]
```

### EVPN, L2VPN, VPLS

```
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
    bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name
    neighbor neighbor-id oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam
    bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)neighbor neighbor-id
    oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group
    mesh-group-name neighbor neighbor-id oam bfd-liveness-detection],
```

```
[edit routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) mesh-group mesh-group-name neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) mesh-group mesh-group-name neighbor
neighbor-id oam bfd-liveness-detection],
```

### Release Information

Statement introduced in Junos OS Release 8.2.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Support for BFD authentication introduced in Junos OS Release 9.6.

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPNs and VPLS.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Enable BFD failure detection. The BFD failure detection timers are adaptive and can be adjusted to be faster or slower. The lower the BFD failure detection timer value, the faster the failure detection and vice versa. For example, the timers can adapt to a higher value if the adjacency fails (that is, the timer detects failures more slowly). Or a neighbor can negotiate a higher value for a timer than the configured value. The timers adapt to a higher value when a BFD session flap occurs more than three times in a span of 15 seconds. A back-off algorithm increases the receive (Rx) interval by two if the local BFD instance is the reason for the session flap. The transmission (Tx) interval is increased by two if the remote BFD instance is the reason for the session flap. You can use the **clear bfd adaptation** command to return BFD interval timers to their configured values. The **clear bfd adaptation** command is hitless, meaning that the command does not affect traffic flow on the routing device.

### Options

**threshold *milliseconds***—Specify the threshold for the adaptation of the BFD session detection time. When the detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

**NOTE:** The threshold value must be equal to or greater than the transmit interval.

The threshold time must be equal to or greater than the value specified in the **minimum-interval** or the **minimum-receive-interval** statement.

**Range:** 1 through 255,000 milliseconds

**Required Privilege Level**

routing—To view this statement in the configuration.

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

**RELATED DOCUMENTATION**

---

[Configuring BFD for Layer 2 VPN and VPLS | 56](#)

---

*Example: Configuring BFD for BGP*

---

*Example: Configuring BFD for Static Routes for Faster Network Failure Detection*

---

*bfd-liveness-detection*



## egress-protection (Layer 2 circuit)

### Syntax

```
egress-protection {
  protected-l2circuit {
    egress-pe address;
    ingress-pe address;
    virtual-circuit-id identifier;
  }
  protector-interface interface-name;
  protector-pe address {
    context-identifier identifier;
    lsp lsp-name;
  }
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name]
```

### Release Information

Statement introduced in Junos OS release 10.4.

### Description

Configures an egress protection virtual circuit (EPVC).

### Options

The other statements are explained separately.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

Example: Configuring an Egress Protection LSP for a Layer 2 Circuit

## egress-protection (MPLS)

### Syntax

```
egress-protection {
  context-identifier context-id {
    primary | protector;
    metric igp-metric-value;
    advertise-mode (stub-alias | stub-proxy);
  }
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols mpls],
[edit logical-systems logical-system-name protocols mpls label-switched-path lsp-name],
[edit protocols mpls],
[edit protocols mpls label-switched-path lsp-name]
```

### Release Information

Statement introduced in Junos OS Release 10.4.

Options **primary**, **protector**, and **metric** introduced in Junos OS Release 11.4R3.

Option **advertise-mode** introduced in Junos OS Release 13.3.

### Description

Enables an Edge Protection Virtual Circuit (EPVC) for the MPLS protocol.

### Options

**context-identifier** *context-id-ip-address*—(Optional) The context identifier IPv4 address.

**metric** *igp-metric-value*—(Optional) The IGP metric value ranging from **2** through **16777215**.

**(primary | protector)**—On the primary PE router, configure as type **primary**. On the protector PE router, configure as type **protector**.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Example: Configuring Egress Protection for Layer 3 VPN Services*

| *Example: Configuring Layer 3 VPN Egress Protection with RSVP and LDP*

## encapsulation (Logical Interface)

### Syntax

```
encapsulation (atm-ccc-cell-relay | atm-ccc-vc-mux | atm-cisco-nlpid | atm-mlppp-llc | atm-nlpid | atm-ppp-llc |
atm-ppp-vc-mux | atm-snap | atm-tcc-snap | atm-tcc-vc-mux | atm-vc-mux | ether-over-atm-llc |
ether-vpls-over-atm-llc | ether-vpls-over-fr | ether-vpls-over-ppp | ethernet | ethernet-ccc | ethernet-vpls |
ethernet-vpls-fr | frame-relay-ccc | frame-relay-ether-type | frame-relay-ether-type-tcc | frame-relay-ppp |
frame-relay-tcc | gre-fragmentation | multilink-frame-relay-end-to-end | multilink-ppp | ppp-over-ether |
ppp-over-ether-over-atm-llc | vlan-bridge | vlan-ccc | vlan-vci-ccc | vlan-tcc | vlan-vpls | vxlan);
```

### Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number],
[edit interfaces rlsq number unit logical-unit-number]
[edit protocols evpn]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers

(**ethernet**, **vlan-ccc**, and **vlan-tcc** options only).

Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro Routers. Only the **atm-ccc-cell-relay** and **atm-ccc-vc-mux** options are supported on ACX Series routers.

Statement introduced in Junos OS Release 17.3R1 for QFX10000 Series switches (**ethernet-ccc** and **vlan-ccc** options only).

### Description

Configure a logical link-layer encapsulation type. Not all encapsulation types are supported on the switches. See the switch CLI.

Starting in Junos OS Release 20.1R1, aggregated ethernet interfaces supports VLAN TCC (Translational cross-connect) encapsulation on MX series platforms. See *Configuring VLAN TCC Encapsulation* for more details. Non-ethernet media types, SONET and ATM interfaces are only supported. It is expected that the user will have the member links of aggregated ethernet with supported hardware for configuring VLAN TCC encapsulation and no commit check is performed externally for the aggregated ethernet (AE) interfaces.

### Options

**atm-ccc-cell-relay**—Use ATM cell-relay encapsulation.

**atm-ccc-vc-mux**—Use ATM virtual circuit (VC) multiplex encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

**atm-cisco-nlpid**—Use Cisco ATM network layer protocol identifier (NLPID) encapsulation. When you use this encapsulation type, you can configure the **inet** family only.

**atm-mlppp-llc**—For ATM2 IQ interfaces only, use Multilink Point-to-Point (MLPPP) over AAL5 LLC. For this encapsulation type, your router must be equipped with a Link Services or Voice Services PIC. MLPPP over ATM encapsulation is not supported on ATM2 IQ OC48 interfaces.

**atm-nlpid**—Use ATM NLPID encapsulation. When you use this encapsulation type, you can configure the **inet** family only.

**atm-ppp-llc**—(ATM2 IQ interfaces and MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP only) Use PPP over AAL5 LLC encapsulation.

**atm-ppp-vc-mux**—(ATM2 IQ interfaces and MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP only) Use PPP over ATM AAL5 multiplex encapsulation.

**atm-snap**—(All interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) Use ATM subnetwork attachment point (SNAP) encapsulation.

**atm-tcc-snap**—Use ATM SNAP encapsulation on translational cross-connect (TCC) circuits.

**atm-tcc-vc-mux**—Use ATM VC multiplex encapsulation on TCC circuits. When you use this encapsulation type, you can configure the **tcc** family only.

**atm-vc-mux**—(All interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) Use ATM VC multiplex encapsulation. When you use this encapsulation type, you can configure the **inet** family only.

**ether-over-atm-llc**—(All IP interfaces including MX Series routers with MPC/MIC interfaces using the ATM MIC with SFP) For interfaces that carry IP traffic, use Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure multipoint interfaces.

**ether-vpls-over-atm-llc**—For ATM2 IQ interfaces only, use the Ethernet virtual private LAN service (VPLS) over ATM LLC encapsulation to bridge Ethernet interfaces and ATM interfaces over a VPLS routing instance (as described in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*). Packets from the ATM interfaces are converted to standard ENET2/802.3 encapsulated Ethernet frames with the frame check sequence (FCS) field removed.

**ether-vpls-over-fr**—For E1, T1, E3, T3, and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over Frame Relay encapsulation to support Bridged Ethernet over Frame Relay encapsulated TDM interfaces for VPLS applications, per RFC 2427, *Multiprotocol Interconnect over Frame Relay*.

**NOTE:** The SONET/SDH OC3/STM1 (Multi-Rate) MIC with SFP, the Channelized SONET/SDH OC3/STM1 (Multi-Rate) MIC with SFP, and the DS3/E3 MIC do not support Ethernet over Frame Relay encapsulation.

**ether-vpls-over-ppp**—For E1, T1, E3, T3, and SONET interfaces only, use the Ethernet virtual private LAN service (VPLS) over Point-to-Point Protocol (PPP) encapsulation to support Bridged Ethernet over PPP-encapsulated TDM interfaces for VPLS applications.

**ethernet**—Use Ethernet II encapsulation (as described in RFC 894, *A Standard for the Transmission of IP Datagrams over Ethernet Networks*).

**ethernet-ccc**—Use Ethernet CCC encapsulation on Ethernet interfaces.

**ethernet-vpls**—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard Tag Protocol ID (TPID) values.

**NOTE:** The built-in Gigabit Ethernet PIC on an M7i router does not support extended VLAN VPLS encapsulation.

**ethernet-vpls-fr**—Use in a VPLS setup when a CE device is connected to a PE router over a time-division multiplexing (TDM) link. This encapsulation type enables the PE router to terminate the outer layer 2 Frame Relay connection, use the 802.1p bits inside the inner Ethernet header to classify the packets, look at the MAC address from the Ethernet header, and use the MAC address to forward the packet into a given VPLS instance.

**frame-relay-ccc**—Use Frame Relay encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

**frame-relay-ether-type**—Use Frame Relay ether type encapsulation for compatibility with Cisco Frame Relay. The physical interface must be configured with flexible-frame-relay encapsulation.

**frame-relay-ether-type-tcc**—Use Frame Relay ether type TCC for Cisco-compatible Frame Relay on TCC circuits to connect different media. The physical interface must be configured with flexible-frame-relay encapsulation.

**frame-relay-ppp**—Use PPP over Frame Relay circuits. When you use this encapsulation type, you can configure the **ppp** family only.

**frame-relay-tcc**—Use Frame Relay encapsulation on TCC circuits for connecting different media. When you use this encapsulation type, you can configure the **tcc** family only.

**gre-fragmentation**—For adaptive services interfaces only, use GRE fragmentation encapsulation to enable fragmentation of IPv4 packets in GRE tunnels. This encapsulation clears the do not fragment (DF) bit in the packet header. If the packet's size exceeds the tunnel's maximum transmission unit (MTU) value, the packet is fragmented before encapsulation.

**multilink-frame-relay-end-to-end**—Use MLFR FRF.15 encapsulation. This encapsulation is used only on multilink, link services, and voice services interfaces and their constituent T1 or E1 interfaces, and is supported on LSQ and redundant LSQ interfaces.

**multilink-ppp**—Use MLPPP encapsulation. This encapsulation is used only on multilink, link services, and voice services interfaces and their constituent T1 or E1 interfaces.

**ppp-over-ether**—Use PPP over Ethernet encapsulation to configure an underlying Ethernet interface for a dynamic PPPoE logical interface on M120 and M320 routers with Intelligent Queuing 2 (IQ2) PICs, and on MX Series routers with MPCs.

**ppp-over-ether-over-atm-llc**—(MX Series routers with MPCs using the ATM MIC with SFP only) For underlying ATM interfaces, use PPP over Ethernet over ATM LLC encapsulation. When you use this encapsulation type, you cannot configure the interface address. Instead, configure the interface address on the PPP interface.

**vlan-bridge**—Use Ethernet VLAN bridge encapsulation on Ethernet interfaces that have IEEE 802.1Q tagging, flexible-ethernet-services, and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID.

**vlan-ccc**—Use Ethernet virtual LAN (VLAN) encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

**vlan-vci-ccc**—Use ATM-to-Ethernet interworking encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.

**vlan-tcc**—Use Ethernet VLAN encapsulation on TCC circuits. When you use this encapsulation type, you can configure the **tcc** family only.

**vlan-vpls**—Use Ethernet VLAN encapsulation on VPLS circuits.

**vxlan**—Use VXLAN data plane encapsulation for EVPN.

#### Required Privilege Level

interface—To view this statement in the configuration.

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

#### Release History Table

Release	Description
<a href="#">20.1R1</a>	Starting in Junos OS Release 20.1R1, aggregated ethernet interfaces supports VLAN TCC (Translational cross-connect) encapsulation on MX series platforms.

## RELATED DOCUMENTATION

*Configuring Layer 2 Switching Cross-Connects Using CCC*

---

*Configuring the Encapsulation for Layer 2 Switching TCCs*

---

*Configuring Interface Encapsulation on Logical Interfaces*

---

*Configuring the CCC Encapsulation for LSP Tunnel Cross-Connects*

---

*Circuit and Translational Cross-Connects Overview*

---

*Identifying the Access Concentrator*

---

*Configuring ATM Interface Encapsulation*

---

*Configuring VLAN and Extended VLAN Encapsulation*

---

*Configuring ATM-to-Ethernet Interworking*

---

*Configuring Interface Encapsulation on PTX Series Packet Transport Routers*

---

[Configuring CCC Encapsulation for Layer 2 VPNs | 34](#)

---

[Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits | 35](#)

---

*Configuring ATM for Subscriber Access*

---

*Understanding CoS on ATM IMA Pseudowire Interfaces Overview*

---

*Configuring Policing on an ATM IMA Pseudowire*



## encapsulation

### List of Syntax

[Syntax for Physical Interfaces: M Series, MX Series, QFX Series, T Series, PTX Series on page 177](#)

[Syntax for Physical Interfaces: SRX Series on page 177](#)

[Syntax for Logical Interfaces: SRX Series on page 177](#)

### Syntax for Physical Interfaces: M Series, MX Series, QFX Series, T Series, PTX Series

```
encapsulation (atm-ccc-cell-relay | atm-pvc | cisco-hdlc | cisco-hdlc-ccc | cisco-hdlc-tcc | ethernet-bridge | ethernet-ccc
| ethernet-over-atm | ethernet-tcc | ethernet-vpls | ethernet-vpls-fr | ether-vpls-over-atm-llc | ethernet-vpls-ppp
| extended-frame-relay-ccc | extended-frame-relay-ether-type-tcc | extended-frame-relay-tcc |
extended-vlan-bridge | extended-vlan-ccc | extended-vlan-tcc | extended-vlan-vpls | flexible-ethernet-services |
flexible-frame-relay | frame-relay | frame-relay-ccc | frame-relay-ether-type | frame-relay-ether-type-tcc |
frame-relay-port-ccc | frame-relay-tcc | generic-services | multilink-frame-relay-uni-nni | ppp | ppp-ccc | ppp-tcc
| vlan-ccc | vlan-vci-ccc | vlan-vpls);
```

### Syntax for Physical Interfaces: SRX Series

```
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 | flexible-ethernet-services | frame-relay-port-ccc | vlan-ccc | vlan-vpls);
```

### Syntax for Logical Interfaces: SRX Series

```
encapsulation ( dix | ether-vpls-fr | frame-relay-ppp | ppp-over-ether | vlan-bridge | vlan-ccc | vlan-tcc | vlan-vpls );
```

### Physical Interfaces: M Series, MX Series, QFX Series, T Series, PTX Series

```
[edit interfaces interface-name],
[edit interfaces rlsq number:number]
```

### Logical Interfaces

```
[edit interfaces interface-name unit logical-unit-number ]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.5.


Statement introduced in Junos OS Release 11.1 for EX Series switches.

Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers (**flexible-ethernet-services**, **ethernet-ccc**, and **ethernet-tcc** options only).

### Description

For M Series, MX Series, QFX Series, T Series, PTX Series, specify the physical link-layer encapsulation type.

For SRX Series, specify logical link layer encapsulation.



**NOTE:** Not all encapsulation types are supported on the switches. See the switch CLI.

### Default

**ppp**—Use serial PPP encapsulation.

## Physical Interface Options and Logical Interface Options

[Warning: element unresolved in stylesheets: <title> (in <config-options>). This is probably a new element that is not yet supported in the stylesheets.]

### Physical Interface Options and Logical Interface Options

For physical interfaces:

**NOTE:** Frame Relay, ATM, PPP, SONET, and SATSOP options are not supported on EX Series switches.

- **atm-ccc-cell-relay**—Use ATM cell-relay encapsulation.
- **atm-pvc**—Defined in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*. When you configure physical ATM interfaces with ATM PVC encapsulation, an RFC 2684-compliant ATM Adaptation Layer 5 (AAL5) tunnel is set up to route the ATM cells over a Multiprotocol Label Switching (MPLS) path that is typically established between two MPLS-capable routers using the Label Distribution Protocol (LDP).
- **cisco-hdlc**—Use Cisco-compatible High-Level Data Link Control (HDLC) framing. E1, E3, SONET/SDH, T1, and T3 interfaces can use Cisco HDLC encapsulation. Two related versions are supported:
  - CCC version (**cisco-hdlc-ccc**)—The logical interface does not require an encapsulation statement. When you use this encapsulation type, you can configure the **ccc** family only.
  - TCC version (**cisco-hdlc-tcc**)—Similar to CCC and has the same configuration restrictions, but used for circuits with different media on either side of the connection.
- **cisco-hdlc-ccc**—Use Cisco-compatible HDLC framing on CCC circuits.
- **cisco-hdlc-tcc**—Use Cisco-compatible HDLC framing on TCC circuits for connecting different media.
- **ethernet-bridge**—Use Ethernet bridge encapsulation on Ethernet interfaces that have bridging enabled and that must accept all packets.
- **ethernet-over-atm**—For interfaces that carry IPv4 traffic, use Ethernet over ATM encapsulation. When you use this encapsulation type, you cannot configure multipoint interfaces. As defined in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*, this encapsulation type allows ATM interfaces to connect to devices that support only bridge protocol data units (BPDUs). Junos OS does not completely support bridging, but accepts BPDU packets as a default gateway. If you use the router as an edge device, then the router acts as a default gateway. It accepts Ethernet LLC/SNAP frames with IP or ARP in the payload, and drops the rest. For packets destined to the Ethernet LAN, a route lookup is done using the destination IP address. If the route lookup yields a full address match, the packet is encapsulated with an LLC/SNAP and MAC header, and the packet is forwarded to the ATM interface.
- **ethernet-tcc**—For interfaces that carry IPv4 traffic, use Ethernet TCC encapsulation on interfaces that must accept packets carrying standard TPID values. For 8-port, 12-port, and 48-port Fast Ethernet PICs, TCC is not supported.

- **ethernet-vpls**—Use Ethernet VPLS encapsulation on Ethernet interfaces that have VPLS enabled and that must accept packets carrying standard TPID values. On M Series routers, except the M320 router, the 4-port Fast Ethernet TX PIC and the 1-port, 2-port, and 4-port, 4-slot Gigabit Ethernet PICs can use the Ethernet VPLS encapsulation type.
- **ethernet-vpls-fr**—Use in a VPLS setup when a CE device is connected to a PE device over a time division multiplexing (TDM) link. This encapsulation type enables the PE device to terminate the outer Layer 2 Frame Relay connection, use the 802.1p bits inside the inner Ethernet header to classify the packets, look at the MAC address from the Ethernet header, and use the MAC address to forward the packet into a given VPLS instance.
- **ethernet-vpls-ppp**—Use in a VPLS setup when a CE device is connected to a PE device over a time division multiplexing (TDM) link. This encapsulation type enables the PE device to terminate the outer Layer 2 PPP connection, use the 802.1p bits inside the inner Ethernet header to classify the packets, look at the MAC address from the Ethernet header, and use it to forward the packet into a given VPLS instance.
- **ether-vpls-over-atm-llc**—For ATM intelligent queuing (IQ) interfaces only, use the Ethernet virtual private LAN service (VPLS) over ATM LLC encapsulation to bridge Ethernet interfaces and ATM interfaces over a VPLS routing instance (as described in RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*). Packets from the ATM interfaces are converted to standard ENET2/802.3 encapsulated Ethernet frames with the frame check sequence (FCS) field removed.
- **extended-frame-relay-ccc**—Use Frame Relay encapsulation on CCC circuits. This encapsulation type allows you to dedicate DLCIs 1 through 1022 to CCC. When you use this encapsulation type, you can configure the **ccc** family only.
- **extended-frame-relay-ether-type-tcc**—Use extended Frame Relay ether type TCC for Cisco-compatible Frame Relay for DLCIs 1 through 1022. This encapsulation type is used for circuits with different media on either side of the connection.
- **extended-frame-relay-tcc**—Use Frame Relay encapsulation on TCC circuits to connect different media. This encapsulation type allows you to dedicate DLCIs 1 through 1022 to TCC.
- **extended-vlan-bridge**—Use extended VLAN bridge encapsulation on Ethernet interfaces that have IEEE 802.1Q VLAN tagging and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID.
- **extended-vlan-ccc**—Use extended VLAN encapsulation on CCC circuits with Gigabit Ethernet and 4-port Fast Ethernet interfaces that must accept packets carrying 802.1Q values. Extended VLAN CCC encapsulation supports TPIDs 0x8100, 0x9100, and 0x9901. When you use this encapsulation type, you can configure the **ccc** family only. For 8-port, 12-port, and 48-port Fast Ethernet PICs, extended VLAN CCC is not supported. For 4-port Gigabit Ethernet PICs, extended VLAN CCC is not supported.
- **extended-vlan-tcc**—For interfaces that carry IPv4 traffic, use extended VLAN encapsulation on TCC circuits with Gigabit Ethernet interfaces on which you want to use 802.1Q tagging. For 4-port Gigabit Ethernet PICs, extended VLAN TCC is not supported.

- **extended-vlan-vpls**—Use extended VLAN VPLS encapsulation on Ethernet interfaces that have VLAN 802.1Q tagging and VPLS enabled and that must accept packets carrying TPIDs 0x8100, 0x9100, and 0x9901. On M Series routers, except the M320 router, the 4-port Fast Ethernet TX PIC and the 1-port, 2-port, and 4-port, 4-slot Gigabit Ethernet PICs can use the Ethernet VPLS encapsulation type.

**NOTE:** The built-in Gigabit Ethernet PIC on an M7i router does not support extended VLAN VPLS encapsulation.

- **flexible-ethernet-services**—For Gigabit Ethernet IQ interfaces and Gigabit Ethernet PICs with small form-factor pluggable transceivers (SFPs) (except the 10-port Gigabit Ethernet PIC and the built-in Gigabit Ethernet port on the M7i router), and for Gigabit Ethernet interfaces, use flexible Ethernet services encapsulation when you want to configure multiple per-unit Ethernet encapsulations. Aggregated Ethernet bundles can use this encapsulation type. This encapsulation type allows you to configure any combination of route, TCC, CCC, Layer 2 virtual private networks (VPNs), and VPLS encapsulations on a single physical port. If you configure flexible Ethernet services encapsulation on the physical interface, VLAN IDs from 1 through 511 are no longer reserved for normal VLANs.
- **flexible-frame-relay**—For IQ interfaces only, use flexible Frame Relay encapsulation when you want to configure multiple per-unit Frame Relay encapsulations. This encapsulation type allows you to configure any combination of TCC, CCC, and standard Frame Relay encapsulations on a single physical port. Also, each logical interface can have any DLCI value from 1 through 1022.
- **frame-relay**—Use Frame Relay encapsulation is defined in RFC 1490, *Multiprotocol Interconnect over Frame Relay*. E1, E3, link services, SONET/SDH, T1, T3, and voice services interfaces can use Frame Relay encapsulation.
- **frame-relay-ccc**—Use Frame Relay encapsulation on CCC circuits. This encapsulation is same as standard Frame Relay for DLCIs 0 through 511. DLCIs 512 through 1022 are dedicated to CCC. The logical interface must also have **frame-relay-ccc** encapsulation. When you use this encapsulation type, you can configure the **ccc** family only.
- **frame-relay-ether-type**—Use Frame Relay ether type encapsulation for compatibility with the Cisco Frame Relay. IETF frame relay encapsulation identifies the payload format using NLPID and SNAP formats. Cisco-compatible Frame Relay encapsulation uses the Ethernet type to identify the type of payload.

**NOTE:** When the encapsulation type is set to Cisco-compatible Frame Relay encapsulation, ensure that the LMI type is set to ANSI or Q933-A.

- **frame-relay-ether-type-tcc**—Use Frame Relay ether type TCC for Cisco-compatible Frame Relay on TCC circuits to connect different media. This encapsulation is Cisco-compatible Frame Relay for DLCIs 0 through 511. DLCIs 512 through 1022 are dedicated to TCC.

- **frame-relay-port-ccc**—Use Frame Relay port CCC encapsulation to transparently carry all the DLCIs between two customer edge (CE) routers without explicitly configuring each DLCI on the two provider edge (PE) routers with Frame Relay transport. The connection between the two CE routers can be either user-to-network interface (UNI) or network-to-network interface (NNI); this is completely transparent to the PE routers. When you use this encapsulation type, you can configure the **ccc** family only.
- **frame-relay-tcc**—This encapsulation is similar to Frame Relay CCC and has the same configuration restrictions, but used for circuits with different media on either side of the connection.
- **generic-services**—Use generic services encapsulation for services with a hierarchical scheduler.
- **multilink-frame-relay-uni-nni**—Use MLFR UNI NNI encapsulation. This encapsulation is used on link services, voice services interfaces functioning as FRF.16 bundles, and their constituent T1 or E1 interfaces, and is supported on LSQ and redundant LSQ interfaces.
- 
- **ppp**—Use serial PPP encapsulation. This encapsulation is defined in RFC 1661, *The Point-to-Point Protocol (PPP) for the Transmission of Multiprotocol Datagrams over Point-to-Point Links*. PPP is the default encapsulation type for physical interfaces. E1, E3, SONET/SDH, T1, and T3 interfaces can use PPP encapsulation.
- **ppp-ccc**—Use serial PPP encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only.
- **ppp-tcc**—Use serial PPP encapsulation on TCC circuits for connecting different media. When you use this encapsulation type, you can configure the **tcc** family only.
- **vlan-ccc**—Use Ethernet VLAN encapsulation on CCC circuits. VLAN CCC encapsulation supports TPID 0x8100 only. When you use this encapsulation type, you can configure the **ccc** family only.
- **vlan-vci-ccc**—Use ATM-to-Ethernet interworking encapsulation on CCC circuits. When you use this encapsulation type, you can configure the **ccc** family only. All logical interfaces configured on the Ethernet interface must also have the encapsulation type set to **vlan-vci-ccc**.
- **vlan-vpls**—Use VLAN VPLS encapsulation on Ethernet interfaces with VLAN tagging and VPLS enabled. Interfaces with VLAN VPLS encapsulation accept packets carrying standard TPID values only. On M Series routers, except the M320 router, the 4-port Fast Ethernet TX PIC and the 1-port, 2-port, and 4-port, 4-slot Gigabit Ethernet PICs can use the Ethernet VPLS encapsulation type.

**NOTE:**

- Label-switched interfaces (LSIs) do not support VLAN VPLS encapsulation. Therefore, you can only use VLAN VPLS encapsulation on a PE-router-to-CE-router interface and not a core-facing interface.
- Starting with Junos OS release 13.3, a commit error occurs when you configure **vlan-vpls** encapsulation on a physical interface and configure **family inet** on one of the logical units. Previously, it was possible to commit this invalid configuration.

For logical interfaces:

- **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*

*Configuring Interface Encapsulation on Physical Interfaces*

[Configuring CCC Encapsulation for Layer 2 VPNs | 34](#)

*Configuring Layer 2 Switching Cross-Connects Using CCC*

[Configuring TCC Encapsulation for Layer 2 VPNs and Layer 2 Circuits | 35](#)

*Configuring ATM Interface Encapsulation*

*Configuring ATM-to-Ethernet Interworking*

*Configuring VLAN and Extended VLAN Encapsulation*

*Configuring VLAN and Extended VLAN Encapsulation*

*Configuring Encapsulation for Layer 2 Wholesale VLAN Interfaces*

[Configuring Interfaces for Layer 2 Circuits | 68](#)

*Configuring Interface Encapsulation on PTX Series Packet Transport Routers*

*Configuring MPLS LSP Tunnel Cross-Connects Using CCC*

*Configuring TCC*

*Configuring VPLS Interface Encapsulation*

*Configuring Interfaces for VPLS Routing*

*Defining the Encapsulation for Switching Cross-Connects*

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## encapsulation-type (Layer 2 Circuits)

### Syntax

```
encapsulation-type (atm-aal5 | atm-cell | atm-cell-port-mode | atm-cell-vc-mode | atm-cell-vp-mode | cesop |
  cisco-hdlc | ethernet | ethernet-vlan | frame-relay | frame-relay-port-mode | interworking | ppp | satop-e1 | satop-e3
  | satop-t1 | satop-t3);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit protocols l2circuit local-switching interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 9.2.

### Description

Specify the type of Layer 2 traffic transiting the Layer 2 circuit.

### Options

**atm-aal5**—ATM Adaptation Layer (AAL/5)

**atm-cell**—ATM cell relay

**atm-cell-port-mode**—ATM cell relay port promiscuous mode

**atm-cell-vc-mode**—ATM VC cell relay nonpromiscuous mode

**atm-cell-vp-mode**—ATM virtual path (VP) cell relay promiscuous mode

**cesop**—CESOP-based Layer 2 circuit

**cisco-hdlc**—Cisco Systems-compatible HDLC

**ethernet**—Ethernet

**ethernet-vlan**—Ethernet VLAN

**frame-relay**—Frame Relay

**frame-relay-port-mode**—Frame Relay port mode

**interworking**—Layer 2.5 interworking

**ppp**—PPP



**satsop-e1**—SATSOP-E1-based Layer 2 circuit

**satsop-e3**—SATSOP-E3-based Layer 2 circuit

**satsop-t1**—SATSOP-T1-based Layer 2 circuit

**satsop-t3**—SATSOP-T3-based Layer 2 circuit

#### **Required Privilege Level**

routing—To view this statement in the configuration.

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

#### **RELATED DOCUMENTATION**

| [Configuring the Encapsulation Type for the Layer 2 Circuit Neighbor Interface](#) | 72

## encapsulation-type (Layer 2 VPNs)

### Syntax

```
encapsulation-type (atm-aal5 | atm-cell | atm-cell-port-mode | atm-cell-vc-mode | atm-cell-vp-mode | cesop |
  cisco-hdlc | ethernet | ethernet-vlan | frame-relay | frame-relay-port-mode | interworking | ppp | satop-e1 | satop-e3
  | satop-t1 | satop-t3);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn neighbor address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls],
[edit protocols l2circuit neighbor address interface interface-name],
[edit routing-instances routing-instance-name protocols l2vpn],
[edit routing-instances routing-instance-name protocols l2vpn neighbor address],
[edit routing-instances routing-instance-name protocols vpls],
[edit routing-instances routing-instance-name protocols vpls neighbor address]
```

### Release Information

Statement introduced in Junos OS Release 9.2.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

### Description

Specify the type of Layer 2 traffic originating from the CE device. Only the **ethernet** and **ethernet-vlan** encapsulation types are supported for VPLS. Not all encapsulation types are supported on the switches. See the switch CLI.

### Options

**atm-aal5**—ATM Adaptation Layer (AAL/5)

**atm-cell**—ATM cell relay

**atm-cell-port-mode**—ATM cell relay port promiscuous mode

**atm-cell-vc-mode**—ATM VC cell relay nonpromiscuous mode

**atm-cell-vp-mode**—ATM virtual path (VP) cell relay promiscuous mode

**cesop**—CESOP-based Layer 2 VPN

**cisco-hdlc**—Cisco Systems-compatible HDLC

**ethernet**—Ethernet

**ethernet-vlan**—Ethernet VLAN

**frame-relay**—Frame Relay

**frame-relay-port-mode**—Frame Relay port mode

**interworking**—Layer 2.5 interworking VPN

**ppp**—PPP

**satsop-e1**—SATSOP-E1-based Layer 2 VPN

**satsop-e3**—SATSOP-E3-based Layer 2 VPN

**satsop-t1**—SATSOP-T1-based Layer 2 VPN

**satsop-t3**—SATSOP-T3-based Layer 2 VPN

**Default:** For VPLS networks, the default encapsulation type is **ethernet**.

#### Required Privilege Level

routing—To view this statement in the configuration.

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

#### RELATED DOCUMENTATION

---

[Configuring the Encapsulation Type | 13](#)

---

*Configuring VPLS Routing Instances*

---

[Configuring the Encapsulation Type for the Layer 2 Circuit Neighbor Interface | 72](#)

---

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## end-interface

### Syntax

```
end-interface {  
    interface interface-name;  
    no-revert;  
    protect-interface interface-name;  
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name],  
[edit protocols l2circuit local-switching interface interface-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Specify the end interface for a local interface switch.

**NOTE:** The protect interface must be configured prior to configuring the **no-revert** statement.

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

| [Configuring Local Interface Switching in Layer 2 Circuits](#) | 65

## family (Protocols BGP)

### Syntax

```
family {
  (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
    (any | flow | labeled-unicast | multicast | unicast | segment-routing-te) {
      accepted-prefix-limit {
        maximum number;
        teardown <percentage-threshold> idle-timeout (forever | minutes);
      }
      add-path {
        receive;
        send {
          include-backup-path backup_path_number;
          multipath;
          path-count number;
          path-selection-mode {
            (all-paths | equal-cost-paths);
          }
          prefix-policy [ policy-names ];
        }
      }
    }
    aigp [disable];
    loops number;
    prefix-limit {
      maximum number;
      teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    protection;
    rib-group group-name;
    topology name {
      community {
        target identifier;
      }
    }
    flow {
      no-install;
      no-validate policy-name;
    }
    labeled-unicast {
      accepted-prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | minutes)>;
      }
    }
  }
}
```

```

aggregate-label {
    community community-name;
}
explicit-null {
    connected-only;
}
prefix-limit {
    maximum number;
    teardown <percentage> <idle-timeout (forever | minutes)>;
}
resolve-vpn;
rib (inet.3 | inet6.3);
rib-group group-name;
traffic-statistics {
    file filename <world-readable | no-world-readable>;
    interval seconds;
}
}
}
route-target {
    accepted-prefix-limit {
        maximum number;
        proxy-generate <route-target-policy route-target-policy-name>;
        teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    advertise-default;
    external-paths number;
    prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | minutes)>;
    }
}
}

```

```

(evpn | inet-mdt | inet-mvpn | inet6-mvpn | l2vpn) {
  signaling {
    accepted-prefix-limit {
      maximum number;
      teardown <percentage-threshold> idle-timeout (forever | minutes);
    }
    add-path {
      receive;
      send {
        include-backup-path backup_path_number;
        multipath;
        path-count number;
        path-selection-mode {
          (all-paths | equal-cost-paths);
        }
        prefix-policy [ policy-names ];
      }
    }
    aigp [disable];
    damping;
    loops number;
    prefix-limit {
      maximum number;
      teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    rib-group group-name;
  }
}
traffic-engineering;
}

```

## Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp],
[edit logical-systems logical-system-name protocols bgp group group-name],
[edit logical-systems logical-system-name protocols bgp group group-name neighbor address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols bgp],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols bgp group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols bgp group group-name
  neighbor address],
[edit protocols bgp],
[edit protocols bgp group group-name],
[edit protocols bgp group group-name neighbor address],
[edit routing-instances routing-instance-name protocols bgp],
[edit routing-instances routing-instance-name protocols bgp group group-name],
[edit routing-instances routing-instance-name protocols bgp group group-name neighbor address]
```

## Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Statement introduced in Junos OS Release 14.1X53-D30 for the QFX Series.

Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

**inet-mvpn** and **inet6-mvpn** statements introduced in Junos OS Release 8.4.

**inet-mdt** statement introduced in Junos OS Release 9.4.

Support for the **loops** statement introduced in Junos OS Release 9.6.

**evpn** statement introduced in Junos OS Release 13.2.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

**traffic-engineering** statement introduced in Junos OS Release 14.2.

**segment-routing-te** option introduced in Junos OS Release 17.4R1 for QFX Series, MX Series, and PTX Series with FPC-PTX-P1-A.

## Description

Enable multiprotocol BGP (MP-BGP) by configuring BGP to carry network layer reachability information (NLRI) for address families other than unicast IPv4, to specify MP-BGP to carry NLRI for the IPv6 address family, or to carry NLRI for VPNs.



## Options

**any**—Configure the family type to be both unicast and multicast.

**evpn**—Configure NLRI parameters for Ethernet VPNs (EVPNs).

**inet**—Configure NLRI parameters for IPv4.

**inet6**—Configure NLRI parameters for IPv6.

**inet-mdt**—Configure NLRI parameters for the multicast distribution tree (MDT) subaddress family identifier (SAFI) for IPv4 traffic in Layer 3 VPNs.

**inet-mvpn**—Configure NLRI parameters for IPv4 for multicast VPNs.

**inet6-mvpn**—Configure NLRI parameters for IPv6 for multicast VPNs.

**inet-vpn**—Configure NLRI parameters for IPv4 for Layer 3 VPNs.

**inet6-vpn**—Configure NLRI parameters for IPv6 for Layer 3 VPNs.

**inet6-vpn**—Configure NLRI parameters for IPv6 for Layer 3 VPNs.

**iso-vpn**—Configure NLRI parameters for IS-IS for Layer 3 VPNs.

**l2vpn**—Configure NLRI parameters for IPv4 for MPLS-based Layer 2 VPNs and VPLS.

**labeled-unicast**—Configure the family type to be labeled-unicast. This means that the BGP peers are being used only to carry the unicast routes that are being used by labeled-unicast for resolving the labeled-unicast routes. This statement is supported only with **inet** and **inet6**.

**multicast**—Configure the family type to be multicast. This means that the BGP peers are being used only to carry the unicast routes that are being used by multicast for resolving the multicast routes.

**unicast**—Configure the family type to be unicast. This means that the BGP peers only carry the unicast routes that are being used for unicast forwarding purposes. The default family type is unicast.

**segment-routing-te**—Configure the family type to be segment routing traffic engineering. This means that BGP peers only carry segment routing policies for traffic steering.

The remaining statements are explained separately. Search for a statement in [CLI Explorer](#) or click a linked statement in the Syntax section for details.

## Required Privilege Level

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

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

## RELATED DOCUMENTATION

*Configuring IBGP Sessions Between PE Routers in VPNs*

---

*Understanding Multiprotocol BGP*

---

*autonomous-system*

---

*local-as*

## family multiservice

### Syntax

```
family multiservice {
  destination-mac;
  label-1;
  label-2;
  payload {
    ip {
      layer-3 {
        (source-ip-only | destination-ip-only);
      }
      layer-3-only;
      layer-4;
    }
  }
  source-mac;
  symmetric-hash {
    complement;
  }
}
```

### Hierarchy Level

```
[edit forwarding-options hash-key]
```

### Release Information

Statement introduced in Junos OS Release 8.0.

**ip**, **label-1**, **label-2**, **layer-3-only**, and **payload** options introduced in Junos OS Release 9.4.

**layer-3**, **layer-4**, **source-ip-only**, and **destination-ip-only** options introduced in Junos OS Release 9.5.

**symmetric-hash** and **complement** options introduced in Junos OS Release 9.6.

### Description

Configure load balancing based on Layer 2 media access control information. On MX Series routers, configure VPLS load balancing. On M120 and M320 routers only, configure VPLS load balancing based on MPLS labels and IP information. For IPv4 traffic, only the IP source and destination addresses are included in the hash key. For MPLS and IPv4 traffic, one or two MPLS labels and IPv4 source and destination addresses are included. For MPLS Ethernet pseudowires, only one or two MPLS labels are included in the hash key.

### Options

You can configure one or more options to load-balance using the packet information that you specify.

**destination-mac**—Include the destination-address MAC information in the hash key for Layer 2 load balancing.

**label-1** (M120 and M320 routers only)—Include the first MPLS label in the hash key. Used for including a one-label packet for per-flow load balancing of IPv4 VPLS traffic based on IP information and MPLS labels.

**label-2** (M120 and M320 routers only)—Include the second MPLS label in the hash key. If both **label-1** and **label-2** are specified, the entire first label and the first 16 bits of the second label are hashed.

**payload** (MX Series, M120, and M320 routers only)—Include the packet's IP payload in the hash key.

- **ip** (MX Series, M120, and M320 routers only)—Include the IP address of the IPv4 or IPv6 payload in the hash key.
- **layer-3** (MX Series routers only)—Use this to include Layer 3 information from the packet's IP payload in the hash key.
- **destination-ip-only** (MX Series routers only)—Use this to include only the destination IP address in the payload in the hash key.
- **source-ip-only** (MX Series routers only)—Use this to include only the source IP address in the payload in the hash key.

**NOTE:** You can include either the **source-ip-only** or the **destination-ip-only** statement, not both. They are mutually exclusive.

- **layer-3-only** (M120, and M320 routers only)—Include only the Layer 3 information from the packet's IP payload in the hash key.
- **layer-4** (MX Series routers only)—Include Layer 4 information from the packet's IP payload in the hash key.

**NOTE:** On MX Series routers only, you can configure either Layer 3 or Layer 4 load balancing, or both at the same time.

**NOTE:** On I chip platforms, an unknown Layer 4 header is excluded from load-balance hashing to avoid undesired packet reordering.

**source-mac**—Include the source-address MAC information in the hash key.

**symmetric-hash** (MX Series routers only)—Configure the symmetric hash or symmetric hash complement for configuring symmetrical load balancing on an 802.3ad Link Aggregation Group.

- **complement** —Include the complement of the symmetric hash in the hash key.

**Required Privilege Level**

interface—To view this statement in the configuration.

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

**RELATED DOCUMENTATION**

---

*Configuring Load Balancing Based on MAC Addresses*

---

*Configuring VPLS Load Balancing Based on IP and MPLS Information*

---

*Configuring VPLS Load Balancing on MX Series 5G Universal Routing Platforms*

---

*Configuring VPLS Load Balancing*

## flow-label-receive-static

### Syntax

```
flow-label-receive-static;
```

### Hierarchy Level

```
[edit protocols l2circuit neighbor neighbor-id interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 14.1.

### Description

Configure the FEC 128 VPWS pseudowire to statically push the flow label on the pseudowire packets sent to the remote egress provider edge (PE) router. The ingress PE router inserts the flow label in the pseudowire packet, irrespective of the information exchanged in the signaling plane. If the egress PE router cannot handle the pseudowire packet marked with the flow label, the packet is dropped.

Flow-aware transport of pseudowires (FAT) flow labels enable load-balancing of MPLS packets across equal-cost multipath (ECMP) paths or link aggregation groups (LAGs) without the need for deep packet inspection of the payload. FAT flow labels can be used for LDP-signaled forwarding equivalence class (FEC) 128 and FEC 129 pseudowires for virtual private LAN service (VPLS) and virtual private wire service (VPWS) networks.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| [Configuring the FAT Flow Label for FEC 128 VPWS Pseudowires for Load-Balancing MPLS Traffic](#)

## flow-label-transmit-static

### Syntax

```
flow-label-transmit-static;
```

### Hierarchy Level

```
[edit protocols l2circuit neighbor neighbor-id interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 14.1.

### Description

Configure the router to statically pop the flow label on the pseudowire packets received from the remote egress provider (PE) router. If the incoming pseudowire packet is not marked with the flow label, the packet is dropped by the egress PE router.

Flow-aware transport of pseudowires (FAT) flow labels enable load-balancing of MPLS packets across equal-cost multipath (ECMP) paths or link aggregation groups (LAGs) without the need for deep packet inspection of the payload. FAT flow labels can be used for LDP-signaled forwarding equivalence class (FEC) 128 and FEC 129 pseudowires for virtual private LAN service (VPLS) and virtual private wire service (VPWS) networks.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| [Configuring the FAT Flow Label for FEC 128 VPWS Pseudowires for Load-Balancing MPLS Traffic](#)

## hot-standby

### Syntax

```
hot-standby;
```

### Hierarchy Level

```
[edit routing-instances routing-instance-name protocols l2vpn site site-name]
```

### Release Information

Statement introduced in Junos OS Release 14.2.

### Description

Turn on the protector behavior for the site. This keeps backup pseudowire in continuous standby mode and ready for traffic forwarding.

### Required Privilege Level

routing—To view this statement in the configuration.

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

### RELATED DOCUMENTATION

Example: Configuring MPLS Egress Protection Service Mirroring for BGP Signaled Layer 2 Services



## hot-standby (Protocols Layer 2 Circuit)

### Syntax

```
hot-standby;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name end-interface
  interface interface-name backup-neighbor address],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name backup-neighbor
  address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor address
  backup-neighbor address],
[edit protocols l2circuit neighbor address interface interface-name backup-neighbor address],
[edit routing-instances routing-instance-name protocols vpls neighbor address backup-neighbor address]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Hierarchy levels associated with the **backup-neighbor** statement added in Junos OS Release 9.2.

### Description

Configure the pseudowire to the specified backup neighbor as the hot-standby. When you configure this statement, traffic flows over both the active and hot-standby pseudowires to the backup device (either a CE device or PE router). The backup device drops the traffic from the hot-standby pseudowire, unless the active pseudowire fails. If the active pseudowire fails, the backup device automatically switches to the hot-standby pseudowire.

Configure the **hot-standby** statement on routers that have both active and standby virtual circuits. Generally, these are access routers. On provider edge routers, configure the **hot-standby-vc-on'** statement to indicate that a hot-standby pseudowire is desired upon arrival of a PW\_FWD\_STDBY status-TLV.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring Layer 2 Circuits over Both RSVP and LDP LSPs | 75](#)

[Configuring Pseudowire Redundancy on the PE Router | 52](#)

[Example: Configuring Layer 2 Circuit Switching Protection | 84](#)

## hot-standby-vc-on (Protocols Layer 2 Circuit)

### Syntax

```
hot-standby-vc-on;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name
  pseudowire-status-tlv],
[edit logical systems logical-system-name routing-instances routing-instance-name protocols vpls mesh-group
  mesh-group-name neighbor address pseudowire-status-tlv],
[edit protocols l2circuit neighbor address interface interface-name pseudowire-status-tlv],
[edit routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name neighbor address
  pseudowire-status-tlv]
```

### Release Information

Statement introduced in Junos OS Release 13.2.

Support for VPLS routing instances added in Junos OS Release 15.1R2.

### Description

On provider edge (PE) aggregation routers, configure the **hot-standby-vc-on** statement to indicate that a hot-standby pseudowire is desired upon arrival of a PW\_FWD\_STDBY status-tlv. This flag indicates the standby state. Configure in conjunction with the **hot-standby** statement on routers that have both active and standby virtual circuits. Generally, these are access routers.

The goal of the **hot-standby** statement is to reduce the amount of traffic being discarded during primary-to-backup transition periods. This statement enables the possibility of keeping both the active and standby paths open within the Layer 2 domain. By having both the active and standby VCs able to send and receive traffic, traffic loops could potentially occur within the Layer 2 domain. In consequence, Layer 2 VCs are kept open only in the PE-to-access direction. In other words, aggregation PE devices can send traffic toward access devices, but access devices send traffic exclusively through the active VC.

In this regard, the **hot-standby** statement is quite similar to the **standby** statement. The **hot-standby** statement allows for a faster forwarding-path switchover during transition periods, as compared to what is allowed by the **standby** statement.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Example: Configuring Pseudowire Redundancy in a Mobile Backhaul Scenario*

---

[hot-standby](#) | **201**

## ignore-encapsulation-mismatch

### Syntax

```
ignore-encapsulation-mismatch;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls mesh-group
  mesh-group-name neighbor neighbor-id],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor neighbor-id],
[edit protocols l2circuit local-switching interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name],
[edit routing-instances routing-instance-name protocols evpn interface interface-name],
[edit routing-instances routing-instance-name protocols vpls mesh-group mesh-group-name neighbor neighbor-id],
[edit routing-instances routing-instance-name protocols vpls neighbor neighbor-id]
```

### Release Information

Statement introduced in Junos OS Release 9.2.

Statement extended to support local switching in Junos OS Release 10.4.

Statement introduced for EVPNs in Junos OS Release 13.2 for MX 3D Series.

### Description

Allow a Layer 2 circuit, VPLS, or EVPN to be established even though the encapsulation configured on the CE device interface does not match the encapsulation configured on the Layer 2 circuit, VPLS, or EVPN interface.

### Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration

## RELATED DOCUMENTATION

*Configuring EVPN Routing Instances*

[Enabling the Layer 2 Circuit When the Encapsulation Does Not Match | 72](#)

## ignore-mtu-mismatch

### Syntax

```
ignore-mtu-mismatch;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn interface
  interface-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls],
[edit protocols l2circuit local-switching interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name],
[edit routing-instances routing-instance-name protocols l2vpn interface interface-name],
[edit routing-instances routing-instance-name protocols vpls]
```

### Release Information

Statement introduced in Junos OS Release 8.5.

Support for remote PE routers added in Junos OS Release 9.2.

Support for Layer 2 VPNs and VPLS added in Junos OS Release 10.4.

### Description

Ignore the MTU configuration set for the physical interface associated with the local switching interface or with the remote PE router. This allows a pseudowire to be brought up between two logical interfaces that are defined on physical interfaces with different MTU values.

### Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration

### RELATED DOCUMENTATION

[Enabling Local Interface Switching When the MTU Does Not Match | 67](#)

[Configuring the MTU for Layer 2 Interfaces | 37](#)

## interface (Protocols Layer 2 Circuit)

### Syntax

```
interface interface-name {
  backup-neighbor address;
  bandwidth (bandwidth | ctnumber bandwidth);
  community community-name;
  connection-protection;
  (control-word | no-control-word);
  description text;
  egress-protection;
  encapsulation-type type;
  flow-label-receive;
  flow-label-receive-static;
  flow-label-transmit;
  flow-label-transmit-static;
  ignore-encapsulation-mismatch;
  ignore-mtu-mismatch;
  mtu mtu-number;
  no-revert;
  oam;
  protect-interface interface-name;
  pseudowire-status-tlv hot-standby-vc-on;
  psn-tunnel-endpoint address;
  revert-time seconds;
  static {
    switchover-delay milliseconds;
    virtual-circuit-id identifier;
  }
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching],
[edit logical-systems logical-system-name protocols l2circuit neighbor address],
[edit protocols l2circuit local-switching],
[edit protocols l2circuit neighbor address]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

**flow-label-receive-static** and **flow-label-transmit-static** options introduced in Junos OS Release 14.1.

### Description

Interface over which Layer 2 circuit traffic travels.

## Options

**interface-name**—Name of the interface to configure.

**NOTE:** The commit operation fails, if the same logical interface is configured for both layer 2 circuit and ccc connection.

**connection-protection**—Enable end-to-end protection through OAM failure detection.

**flow-label-receive**—Advertise capability to pop flow label in receive direction to the remote provider edge (PE) device.

**flow-label-receive-static**—Pop flow label on the pseudowire packets received from the remote PE device. The ingress PE inserts the flow label in the pseudowire packet, irrespective of the information exchanged in the signaling plane. If the egress PE cannot handle the pseudowire packet marked with the flow label, the packet is dropped.

**flow-label-transmit**—Advertise capability to push flow label in transmit direction to the remote PE device.

**flow-label-transmit-static**—Push flow label on the pseudowire packets sent to the remote PE device. If the incoming pseudowire packet is not marked with the flow label, the packet is dropped by the egress PE.

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

| [Configuring the Neighbor Interface for the Layer 2 Circuit](#) | 69

## interface (Protocols Layer 2 VPN)

### Syntax

```
interface interface-name {  
    description text;  
    remote-site-id remote-site-id;  
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn],  
[edit routing-instances routing-instance-name protocols l2vpn]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Configure an interface to handle traffic for a circuit configured for the Layer 2 VPN.

### Options

***interface-name***—Name of the interface used for the Layer 2 VPN.

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

---

[Configuring the Site | 11](#)

[Configuring the Remote Site ID | 12](#)



## install-nexthop

### Syntax

```
install-nexthop (except | lsp lsp-name | lsp-regex lsp-regular-expression);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name policy-options policy-statement policy-name term term-name then],  
[edit policy-options policy-statement policy-name term term-name then]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Select a specific label-switched path (LSP), or select an LSP from a set of similarly named LSPs as the traffic destination for the configured community. Also can prevent the installation of any matching next hops.

### Options

**except**—Prevent the installation of any matching next hops.

**lsp *lsp-name***—Configure a specific LSP.

**lsp-regex *lsp-regular-expression***—Configure a range of similarly named LSPs. You can use the following wildcard characters when configuring an LSP regular expression:

- Asterisk (\*)—Match any characters.
- Period (.)—Match any single digit.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

Configuring the Policy Statement for the Layer 2 Circuit Community | 79

## I2circuit

### Syntax

```

I2circuit {
  auto-sensing{
    password password;
  }
  local-switching {
    interface interface-name {
      description text;
      end-interface {
        interface interface-name;
        protect-interface interface-name;
      }
      ignore-mtu-mismatch;
      protect-interface interface-name;
    }
  }
  neighbor address {
    interface interface-name {
      backup-neighbor address;
      bandwidth (bandwidth | ctnumber bandwidth);
      community community-name;
      connection-protection;
      (control-word | no-control-word);
      description text;
      egress-protection;
      encapsulation-type type;
      ignore-encapsulation-mismatch;
      ignore-mtu-mismatch;
      mtu mtu-number;
      protect-interface interface-name;
      pseudowire-status-tlv hot-standby-vc-on;
      psn-tunnel-endpoint address;
      virtual-circuit-id identifier;
    }
  }
  traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
  }
}

```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols],
[edit protocols]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

Statement introduced in Junos OS Release 14.1X53-D10 for the QFX Series and for EX4600 switches.

### Description

Enables a Layer 2 circuit.

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

*Configuring ATM Trunking on Layer 2 Circuits*

*Configuring Bandwidth Allocation and Call Admission Control in Layer 2 Circuits*

[Configuring Interfaces for Layer 2 Circuits | 68](#)

[Configuring LDP for Layer 2 Circuits | 81](#)

[Configuring Policies for Layer 2 Circuits | 78](#)

[Configuring Static Layer 2 Circuits | 64](#)

[Tracing Layer 2 Circuit Operations | 116](#)

## I2ckt

### Syntax

```
I2ckt {  
    l2vpn;  
    l3vpn;  
    labeled-bgp;  
    no-l2vpn;  
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options forwarding-table chained-composite-next-hop ingress],  
[edit routing-options forwarding-table chained-composite-next-hop ingress]
```

### Release Information

Statement introduced in Junos OS Release 13.3.

### Description

Enable composite chained next hop for ingress Layer 2 circuit label-switched paths (LSPs).

The remaining statements are explained separately.

### Required Privilege Level

routing—To view this statement in the configuration.

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

### RELATED DOCUMENTATION

## I2vpn

### Syntax

```

I2vpn {
  (control-word | no-control-word);
  encapsulation-type type;
  oam {
    bfd-liveness-detection {
      detection-time {
        threshold milliseconds;
      }
      minimum-interval milliseconds;
      minimum-receive-interval milliseconds;
      multiplier number;
      no-adaptation;
      transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
      }
      version (1 | automatic);
    }
    ping-interval seconds;
  }
  site site-name {
    community COMM;
    control-word ;
    encapsulation-type ethernet;
    ignore-encapsulation-mismatch;
    ignore-mtu-mismatch;
    interface interface-name {
      description text;
      community COMM;
      control-word ;
      encapsulation-type ethernet;
      ignore-encapsulation-mismatch;
      ignore-mtu-mismatch;
      mtu 1500;
      no-control-word;
      oam {
        bfd-liveness-detection {
          detection-time {
            threshold milliseconds;
          }
          minimum-interval milliseconds;
        }
      }
    }
  }
}

```

```

    minimum-receive-interval milliseconds;
    multiplier number;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    version (1 | automatic);
}
ping-interval seconds; seconds;
}
remote-site-id remote-site-id;
target-attachment-identifier identifier;
}
mtu 1500;
no-control-word;
oam {
    bfd-liveness-detection {
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        multiplier number;
        no-adaptation;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        version (1 | automatic);
    }
    ping-interval seconds; seconds;
}
site-identifier identifier;
site-preference preference-value {
    backup;
    primary;
}
source-attachment-identifier identifier;
}
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}

```

```
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols],  
[edit routing-instances routing-instance-name protocols]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

### Description

Enable a Layer 2 VPN routing instance on a PE router or switch.

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

[Configuring a Layer 2 VPN Routing Instance | 10](#)

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## I2vpn (routing-options)

### Syntax

```
I2vpn {  
  I2ckt;  
  I3vpn;  
  labeled-bgp;  
  no-I2ckt;  
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options forwarding-table chained-composite-next-hop ingress],  
[edit routing-options forwarding-table chained-composite-next-hop ingress]
```

### Release Information

Statement introduced in Junos OS Release 13.3.

### Description

Enable composite chained next hop for ingress Layer 2 virtual private network (VPN) label-switched paths (LSPs).

The remaining statements are explained separately.

### Required Privilege Level

routing—To view this statement in the configuration.

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

### RELATED DOCUMENTATION



## I2vpn-id

### Syntax

```
I2vpn-id (as-number:id | ip-address:id);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances instance-name],  
[edit routing-instances instance-name]
```

### Release Information

Statement introduced in Junos OS Release 10.4R2.

### Description

Specify a globally unique Layer 2 VPN community identifier for the instance.

### Options

***as-number:id***—Autonomous system number (***I2vpn-id:as-number:2-byte-number***. For example: ***I2vpn-id I2vpn-id:100:200***. The AS number can be in the range from 1 through 65,535.

***ip-address:id***—IP address (***I2vpn-id:ip-address:2-byte-number***. For example: ***I2vpn-id I2vpn-id:10.1.1.1:2***. The IP address can be any globally unique unicast address.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Example: Configuring BGP Autodiscovery for LDP VPLS*

*Example: Configuring BGP Autodiscovery for LDP VPLS with User-Defined Mesh Groups*

*Example: Configuring FEC 129 BGP Autodiscovery for VPWS*

## local-switching (Layer 2 Circuits)

### Syntax

```
local-switching {
  interface interface-name {
    description text;
    encapsulation-type;
    end-interface {
      interface interface-name;
      no-revert;
      protect-interface interface-name;
    }
    ignore-encapsulation-mismatch;
    ignore-mtu-mismatch;
    no-revert;
    protect-interface interface-name;
  }
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit],
[edit protocols l2circuit]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Configure a local switching interface. A local switching interface allows you to terminate a virtual circuit on the local router.

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

[Configuring Local Interface Switching in Layer 2 Circuits](#) | 65

## minimum-interval (BFD Liveness Detection)

### Syntax

```
minimum-interval milliseconds;
```

### Hierarchy Level

```
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name
neighbor neighbor-id oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam
bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)neighbor neighbor-id
oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group
mesh-group-name neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor
neighbor-id oam bfd-liveness-detection],
```

### Release Information

Statement introduced in Junos OS Release 8.5.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPN and VPLS.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Configure the minimum interval after which the local routing device transmits hello packets and then expects to receive a reply from a neighbor with which it has established a BFD session. Optionally, instead of using this statement, you can specify the minimum transmit and receive intervals separately using the **minimum-interval** (specified under the **transmit-interval** statement) and **minimum-receive-interval** statements.

**Options**

*milliseconds*—Specify the minimum interval value for BFD liveness detection.

**Range:** 1 through 255,000

**Required Privilege Level**

routing—To view this statement in the configuration.

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

**RELATED DOCUMENTATION**

[Configuring BFD for Layer 2 VPN and VPLS | 56](#)

*Example: Configuring BFD for Static Routes for Faster Network Failure Detection*

[bfd-liveness-detection | 157](#)

[minimum-receive-interval | 221](#)

[transmit-interval | 270](#)

## minimum-receive-interval (BFD Liveness Detection)

### Syntax

```
minimum-receive-interval milliseconds;
```

### Hierarchy Level

```
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name
neighbor neighbor-id oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam
bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)neighbor neighbor-id
oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group
mesh-group-name neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor
neighbor-id oam bfd-liveness-detection],
```

### Release Information

Statement introduced in Junos OS Release 8.5.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Support for BFD authentication introduced in Junos OS Release 9.6.

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPN and VPLS.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Configure the minimum interval after which the local routing device must receive a reply from a neighbor with which it has established a BFD session. Optionally, instead of using this statement, you can configure the minimum receive interval using the **minimum-interval** statement.

### Options

*milliseconds*—Specify the minimum receive interval value.

**Range:** 1 through 255,000

**Required Privilege Level**

routing—To view this statement in the configuration.

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

RELATED DOCUMENTATION

<a href="#">Configuring BFD for Layer 2 VPN and VPLS   56</a>
<i>Example: Configuring BFD for Static Routes for Faster Network Failure Detection</i>
<a href="#">bfd-liveness-detection   157</a>
<a href="#">minimum-interval   219</a>
<a href="#">transmit-interval   270</a>

## mtu

### Syntax

```
mtu bytes;
```

### Hierarchy Level

```
[edit interfaces interface-name],
[edit interfaces interface-name unit logical-unit-number family family],
[edit interfaces interface-range name],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number family family],
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name backup-neighbor
address],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name backup-neighbor
address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn interface
interface-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls],
[edit protocols l2circuit local-switching interface interface-name backup-neighbor address],
[edit protocols l2circuit neighbor address interface interface-name]
[edit protocols l2circuit neighbor address interface interface-name backup-neighbor address],
[edit routing-instances routing-instance-name protocols l2vpn interface interface-name],
[edit routing-instances routing-instance-name protocols vpls],
[edit logical-systems name protocols ospf area name interface ],
[edit logical-systems name routing-instances name protocols ospf area name interface],
[edit protocols ospf area name interface ],
[edit routing-instances name protocols ospf area name interface]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Support for Layer 2 VPNs and VPLS introduced in Junos OS Release 10.4.

Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.

Statement introduced in Junos OS Release 12.2 for ACX Series Universal Metro Routers.

Support at the **[set interfaces interface-name unit logical-unit-number family ccc]** hierarchy level introduced in Junos OS Release 12.3R3 for MX Series routers.

Statement introduced in Junos OS 17.3R1 Release for MX Series Routers.

### Description

Specify the maximum transmission unit (MTU) size for the media or protocol. The default MTU size depends on the device type. Changing the media MTU or protocol MTU causes an interface to be deleted and added again.

To route jumbo data packets on an integrated routing and bridging (IRB) interface or routed VLAN interface (RVI) on EX Series switches, you must configure the jumbo MTU size on the member physical interfaces of the VLAN that you have associated with the IRB interface or RVI, as well as on the IRB interface or RVI itself (the interface named `irb` or `vlan`, respectively).



**CAUTION:** For EX Series switches, setting or deleting the jumbo MTU size on an IRB interface or RVI while the switch is transmitting packets might cause packets to be dropped.

**NOTE:**

The MTU for an IRB interface is calculated by removing the Ethernet header overhead  $[6(\text{DMAC})+6(\text{SMAC})+2(\text{EtherType})]$ . Because, the MTU is the lower value of the MTU configured on the IRB interface and the MTU configured on the IRB's associated bridge domain IFDs or IFLs, the IRB MTU is calculated as follows:

- In case of Layer 2 IFL configured with the **flexible-vlan-tagging** statement, the IRB MTU is calculated by including 8 bytes overhead (SVLAN+CVLAN).
- In case of Layer 2 IFL configured with the **vlan-tagging** statement, the IRB MTU is calculated by including a single VLAN 4 bytes overhead.



**NOTE:**

- If a packet whose size is larger than the configured MTU size is received on the receiving interface, the packet is eventually dropped. The value considered for MRU (maximum receive unit) size is also the same as the MTU size configured on that interface.
- Not all devices allow you to set an MTU value, and some devices have restrictions on the range of allowable MTU values. You cannot configure an MTU for management Ethernet interfaces (fxp0, em0, or me0) or for loopback, multilink, and multicast tunnel devices.
- On ACX Series routers, you can configure the protocol MTU by including the **mtu** statement at the **[edit interfaces interface-name unit logical-unit-number family inet]** or **[edit interfaces interface-name unit logical-unit-number family inet6]** hierarchy level.
  - If you configure the protocol MTU at any of these hierarchy levels, the configured value is applied to all families that are configured on the logical interface.
  - If you are configuring the protocol MTU for both **inet** and **inet6** families on the same logical interface, you must configure the same value for both the families. It is not recommended to configure different MTU size values for **inet** and **inet6** families that are configured on the same logical interface.
- Starting in Release 14.2, MTU for IRB interfaces is calculated by removing the Ethernet header overhead (**6(DMAC)+6(SMAC)+2(EtherType)**), and the MTU is a minimum of the two values:
  - Configured MTU
  - Associated bridge domain's physical or logical interface MTU
    - For Layer 2 logical interfaces configured with **flexible-vlan-tagging**, IRB MTU is calculated by including 8 bytes overhead (**SVLAN+CVLAN**).
    - For Layer 2 logical interfaces configured with **vlan-tagging**, IRB MTU is calculated by including single VLAN 4 bytes overhead.

**NOTE:** Changing the Layer 2 logical interface option from **vlan-tagging** to **flexible-vlan-tagging** or vice versa adjusts the logical interface MTU by 4 bytes with the existing MTU size. As a result, the Layer 2 logical interface is deleted and re-added, and the IRB MTU is re-computed appropriately.

For more information about configuring MTU for specific interfaces and router or switch combinations, see *Configuring the Media MTU*.

### Options

**bytes**—MTU size.

**Range:** 256 through 9192 bytes, 256 through 9216 (EX Series switch interfaces), 256 through 9500 bytes (Junos OS 12.1X48R2 for PTX Series routers), 256 through 9500 bytes (Junos OS 16.1R1 for MX Series routers)

**NOTE:** Starting in Junos OS Release 16.1R1, the MTU size for a media or protocol is increased from 9192 to 9500 for Ethernet interfaces on the following MX Series MPCs:

- MPC1
- MPC2
- MPC2E
- MPC3E
- MPC4E
- MPC5E
- MPC6E

**Default:** 1500 bytes (INET, INET6, and ISO families), 1448 bytes (MPLS), 1514 bytes (EX Series switch interfaces)

### Required Privilege Level

interface—To view this statement in the configuration.

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

### RELATED DOCUMENTATION

*Configuring the Media MTU*

[Configuring the MTU for Layer 2 Interfaces | 37](#)

*Setting the Protocol MTU*

## multiplier (BFD Liveness Detection)

### Syntax

```
multiplier number;
```

### Hierarchy Level

```
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name
neighbor neighbor-id oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam
bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)neighbor neighbor-id
oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group
mesh-group-name neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor
neighbor-id oam bfd-liveness-detection],
```

### Release Information

Statement introduced in Junos OS Release 8.5.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Support for BFD authentication introduced in Junos OS Release 9.6.

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPN and VPLS.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Configure the number of hello packets not received by a neighbor that causes the originating interface to be declared down.

### Options

***number***—Number of hello packets.

**Range:** 1 through 255

**Default:** 3

**Required Privilege Level**

routing—To view this statement in the configuration.

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

**RELATED DOCUMENTATION**

---

[Configuring BFD for Layer 2 VPN and VPLS | 56](#)

---

*Example: Configuring BFD for Static Routes for Faster Network Failure Detection*

---

[bfd-liveness-detection | 157](#)

## neighbor (Protocols Layer 2 Circuit)

### Syntax

```
neighbor address {
  interface interface-name {
    backup-neighbor address {
      community name;
      hot-standby;
      psn-tunnel-endpoint address;
      standby;
      virtual-circuit-id number;
    }
    bandwidth (bandwidth | ctnumber bandwidth);
    community community-name;
    (control-word | no-control-word);
    description text;
    egress-protection {
      protected-l2circuit {
        egress-pe address;
        ingress-pe address;
        virtual-circuit-id identifier;
      }
      protector-interface interface-name;
      protector-pe address {
        context-identifier identifier;
        lsp lsp-name;
      }
    }
  }
  encapsulation-type type;
  ignore-encapsulation-mismatch;
  ignore-mtu-mismatch;
  mtu mtu-number;
  no-revert;
  protect-interface interface-name;
  pseudowire-status-tlv hot-standby-vc-on;
  psn-tunnel-endpoint address;
  revert-time seconds;
  static {
    incoming-label label;
    outgoing-label label;
    send-oam;
  }
  switchover-delay milliseconds;
```

```

    virtual-circuit-id identifier;
  }
}

```

### Hierarchy Level

```

[edit logical-systems logical-system-name protocols l2circuit],
[edit protocols l2circuit]

```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

### Description

Each Layer 2 circuit is represented by the logical interface connecting the local provider edge (PE) router or switch to the local customer edge (CE) router or switch. All the Layer 2 circuits using a particular remote PE router or switch designated for remote CE routers or switches are listed under the **neighbor** statement (neighbor designates the PE router or switch). Each neighbor is identified by its IP address and is usually the end-point destination for the LSP tunnel (transporting the Layer 2 circuit).

### Options

**address**—IP address of a neighboring router or switch.

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

| [Configuring the Neighbor Interface for the Layer 2 Circuit](#) | 69

## no-adaptation (BFD Liveness Detection)

### Syntax

```
no-adaptation;
```

### Hierarchy Level

```
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name
neighbor neighbor-id oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam
bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)neighbor neighbor-id
oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group
mesh-group-name neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor
neighbor-id oam bfd-liveness-detection],
```

### Release Information

Statement introduced in Junos OS Release 9.0

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Support for BFD authentication introduced in Junos OS Release 9.6.

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPN and VPLS.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Configure BFD sessions not to adapt to changing network conditions. We recommend that you *do not* disable BFD adaptation unless it is preferable to have BFD adaptation disabled in your network.

The BFD failure detection timers are adaptive and can be adjusted to be faster or slower. The lower the BFD failure detection timer value, the faster the failure detection and vice versa. For example, the timers can adapt to a higher value if the adjacency fails (that is, the timer detects failures more slowly). Or a

neighbor can negotiate a higher value for a timer than the configured value. The timers adapt to a higher value when a BFD session flap occurs more than three times in a span of 15 seconds. A back-off algorithm increases the receive (Rx) interval by two if the local BFD instance is the reason for the session flap. The transmission (Tx) interval is increased by two if the remote BFD instance is the reason for the session flap. However, include the **no-adaptation** statement in the configuration if you do not want BFD sessions to adapt to changing network conditions.

You can use the **clear bfd adaptation** command to return BFD interval timers to their configured values. The **clear bfd adaptation** command does not affect traffic flow on the routing device.

#### Required Privilege Level

routing—To view this statement in the configuration.

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

#### RELATED DOCUMENTATION

---

[Configuring BFD for Layer 2 VPN and VPLS | 56](#)

---

*Example: Configuring BFD for Static Routes for Faster Network Failure Detection*

---

[bfd-liveness-detection | 157](#)



## no-control-word (Protocols Layer 2 VPN)

### Syntax

```
no-control-word;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn],  
[edit routing-instances routing-instance-name protocols l2vpn]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Disable the control word. This might be necessary on networks with equipment that does not support the control word.

**NOTE:** The following configuration statements are ignored for time-division multiplexing pseudowires at the **[edit protocols l2vpn]** hierarchy level:

- control-word
- no-control-word

### Default

The control word is enabled by default. Use the **no-control-word** statement to disable the control word.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Disabling the Control Word for Layer 2 VPNs](#) | 38

[control-word](#) | 163

## no-l2ckt

### Syntax

```
no-l2ckt;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options forwarding-table chained-composite-next-hop ingress],  
[edit routing-options forwarding-table chained-composite-next-hop ingress]
```

### Release Information

Statement introduced in Junos OS Release 13.3.

### Description

Disable composite chained next hop for ingress Layer 2 circuit label-switched paths (LSPs).

The remaining statements are explained separately.

**NOTE:** For PTX Series routers, it is recommended that you do not use this command and disable composite chained next hop for ingress LSPs.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## no-l2vpn

### Syntax

```
no-l2vpn {  
    l2ckt;  
    l3vpn;  
    labeled-bgp;  
    no-l2ckt;  
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-options forwarding-table chained-composite-next-hop ingress],  
[edit routing-options forwarding-table chained-composite-next-hop ingress]
```

### Release Information

Statement introduced in Junos OS Release 13.3.

### Description

Disable composite chained next hop for ingress Layer 2 virtual private networkk (VPN) label-switched paths (LSPs).

The remaining statements are explained separately.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## no-revert (Local Switching)

### Syntax

```
no-revert;
```

### Hierarchy Level

```
[edit protocols l2circuit local-switching interfaces interface-name]  
[edit protocols l2circuit local-switching interfaces interface-name end-interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 11.4.

### Description

(Optional) Prevent the local switching interface from reverting to the primary interface.

**NOTE:** The protect interface must be configured prior to configuring the **no-revert** statement.

Typically, when the primary interface goes down, the pseudowire starts using the protect interface. By default, when the primary interface comes back online, the interface is switched-over back from the protect interface to the primary interface. To prevent the switchover back to the primary interface, unless the primary interface goes down, include the **no-revert** statement. This prevents loss of traffic during the switchover.

**NOTE:** If the protect interface fails, the interface is switched-over back to the primary interface, irrespective of whether or not the **no-revert** statement is included in the configuration.

This statement can be configured both for the starting interface and the ending interface.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring Local Interface Switching in Layer 2 Circuits](#) | 65

## no-revert (Neighbor Interface)

### Syntax

```
no-revert;
```

### Hierarchy Level

```
[edit protocols l2circuit neighbor address interfaces interface-name],  
[edit logical-systems logical-system-name protocols l2circuit neighbor address interfaces interface-name]
```

### Release Information

Statement introduced in Junos OS Release 11.3.

### Description

(Optional) Prevent the protect interface from reverting to the primary interface.

**NOTE:** The protect interface must be configured prior to configuring the **no-revert** statement.

Typically, when the primary interface goes down, the pseudowire starts using the protect interface. By default, when the primary interface comes back online, the interface is switched-over back from the protect interface to the primary interface. To prevent the switchover back to the primary interface, unless the protect interface goes down, include the **no-revert** statement. This prevents loss of traffic during the switchover.

**NOTE:** If the protect interface fails, the interface is switched-over back to the primary interface, irrespective of whether or not the **no-revert** statement is included in the configuration.

### Required Privilege Level

routing—To view this statement in the configuration.

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

### RELATED DOCUMENTATION

[Configuring Interfaces for Layer 2 Circuits](#) | 68

## oam

### Syntax

```
oam {
    bfd-liveness-detection;
    ping-interval;
    ping-multiplier ping-count;
}
```

### Hierarchy Level

```
[edit routing-instances routing-instance-name protocols l2vpn],
[edit routing-instances routing-instance-name protocols vpls],
[edit routing-instances routing-instance-name protocols vpls neighbor address],
[edit protocols l2circuit neighbor address interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 10.0.

Support for VPLS FEC 129 introduced in Junos OS Release 12.2.

**ping-multiplier** statement introduced in Junos OS Release 16.1.

### Description

Allows you to configure bidirectional forwarding detection (BFD) and a control channel for a pseudowire, in addition to the corresponding operations and management functions to be used over that control channel. BFD provides a low resource fault detection mechanism for the continuous monitoring of the pseudowire data path and for detecting data plane failures. The **control-channel** statement is not applicable to Layer 2 circuit pseudowires.

### Options

**bfd-liveness-detection**—The **bfd-liveness-detection** statement and substatements are described in the *Junos OS Routing Protocols Library*.

**ping-multiplier ping-count**—Specify the number of LSP ping packets to delay the virtual circuit connectivity verification (VCCV) Bidirectional Forwarding Detection (BFD) session from going down. The VCCV BFD session is signaled down only after the specified number of LSP ping packets are lost. This feature is supported for Layer 2 Circuit, Layer 2 VPN, and VPLS technologies.

The other statements are explained separately.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

Configuring BFD for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS | 59

## path-selection

### Syntax

```
path-selection {
  (always-compare-med | cisco-non-deterministic | external-router-id);
  as-path-ignore;
  l2vpn-use-bgp-rules;
  med-plus-igp {
    igp-multiplier number;
    med-multiplier number;
  }
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols bgp],
[edit protocols bgp],
[edit routing-instances routing-instance-name protocols bgp]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Statement introduced in Junos OS Release 11.3 for the QFX Series.

**med-plus-igp** option introduced in Junos OS Release 8.1.

**as-path-ignore** and **l2vpn-use-bgp-rules** options introduced in Junos OS Release 10.2.

Statement introduced in Junos OS Release 12.3 for ACX Series routers.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Configure BGP path selection.

### Default

If the **path-selection** statement is not included in the configuration, only the multiple exit discriminators (MEDs) of routes that have the same peer ASs are compared.

### Options

**always-compare-med**—Always compare MEDs whether or not the peer ASs of the compared routes are the same.



**NOTE:** We recommend that you configure the **always-compare-med** option.

**as-path-ignore**—In the best-path algorithm, skip the step that compares the autonomous system (AS) path lengths. By default, the best-path algorithm evaluates the length of the AS paths and prefers the route with the shortest AS path length.

**NOTE:** Starting with Junos OS Release 14.1R8, 14.2R7, 15.1R4, 15.1F6, and 16.1R1, the **as-path-ignore** option is supported for routing instances.

**cisco-non-deterministic**—Emulate the Cisco IOS default behavior. This mode evaluates routes in the order that they are received and does not group them according to their neighboring AS. With **cisco-non-deterministic** mode, the active path is always first. All inactive, but eligible, paths follow the active path and are maintained in the order in which they were received, with the most recent path first. Ineligible paths remain at the end of the list.

As an example, suppose you have three path advertisements for the 192.168.1.0 /24 route:

- Path 1—learned through EBGp; AS Path of 65010; MED of 200
- Path 2—learned through IBGP; AS Path of 65020; MED of 150; IGP cost of 5
- Path 3—learned through IBGP; AS Path of 65010; MED of 100; IGP cost of 10

These advertisements are received in quick succession, within a second, in the order listed. Path 3 is received most recently, so the routing device compares it against path 2, the next most recent advertisement. The cost to the IBGP peer is better for path 2, so the routing device eliminates path 3 from contention. When comparing paths 1 and 2, the routing device prefers path 1 because it is received from an EBGp peer. This allows the routing device to install path 1 as the active path for the route.

**NOTE:** We do not recommend using this configuration option in your network. It is provided solely for interoperability to allow all routing devices in the network to make consistent route selections.

**external-router-id**—Compare the router ID between external BGP paths to determine the active path.

**igp-multiplier *number***—The multiplier value for the IGP cost to a next-hop address. This option is useful for making the MED and IGP cost comparable.

**Range:** 1 through 1000

**Default:** 1

**l2vpn-use-bgp-rules**—Enable routers to use both the BGP path selection algorithm and the designated forwarder path selection algorithm when selecting the preferred path to each destination in a Layer 2 VPN or VPLS routing instance. The BGP path selection algorithm is used by all of the Provider routers participating in the routing instance. The designated forwarder path selection algorithm is used by the PE router participating in the routing instance.

**Default:** By default, the designated forwarder path selection algorithm is used to select the best path to reach each destination within Layer 2 VPN and VPLS routing instances.

**med-multiplier *number***—The multiplier value for the MED calculation. This option is useful for making the MED and IGP cost comparable.

**Range:** 1 through 1000

**Default:** 1

**med-plus-igp**—Add the IGP cost to the indirect next-hop destination to the MED before comparing MED values for path selection. This statement only affects best-path selection. It does not affect the advertised MED.

#### Required Privilege Level

routing—To view this statement in the configuration.

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

#### RELATED DOCUMENTATION

[Understanding BGP Path Selection | 40](#)

[Enabling BGP Path Selection for Layer 2 VPNs and VPLS | 46](#)

[route-distinguisher | 136](#)

*Example: Ignoring the AS Path Attribute When Selecting the Best Path*

## ping-interval

### Syntax

```
ping-interval seconds;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name oam],
[edit logical-systems logical-system-name routing-instances instance-name protocols l2vpn oam],
[edit logical-systems logical-system-name routing-instances instance-name protocols vpls neighbor address oam],
[edit logical-systems logical-system-name routing-instances instance-name protocols vpls mesh-group mesh-group-name
  neighbor address oam],
[edit logical-systems logical-system-name routing-instances instance-name protocols vpls oam],
[edit protocols l2circuit neighbor address interface interface-name oam],
[edit routing-instances instance-name protocols l2vpn oam],
[edit routing-instances instance-name protocols vpls neighbor address oam],
[edit routing-instances instance-name protocols vpls mesh-group mesh-group-name neighbor address oam],
[edit routing-instances instance-name protocols vpls oam]
```

### Release Information

Statement introduced in Junos OS Release 10.0.

Support for FEC 129 VPLS added in Junos OS Release 12.2.

### Description

Configure the time interval between ping messages for bidirectional forwarding detection (BFD) sessions enabled over pseudowires inside a VPN.

### Options

*seconds*—Time interval between ping messages.

**Range:** 30 through 3600

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring BFD for VCCV for Layer 2 VPNs, Layer 2 Circuits, and VPLS](#) | 59 in the *Junos OS VPNs Library for Routing Devices*

## policer (Layer 2 VPN)

### Syntax

```
policer {
  input policer-template-name;
  output policer-template-name;
}
```

### Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number family (ccc | inet | tcc)],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number family (ccc | inet | tcc)]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Use policing to control the amount of traffic flowing over the interfaces servicing a Layer 2 VPN.

### Options

**input *policer-template-name***—Name of one policer to evaluate when packets are received on the interface.

**output *policer-template-name***—Name of one policer to evaluate when packets are transmitted on the interface.

### Required Privilege Level

interface—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Configuring Traffic Policing in Layer 2 VPNs*

*Routing Policies, Firewall Filters, and Traffic Policers User Guide*

*Junos OS Network Interfaces Library for Routing Devices*

## protect-interface

### Syntax

```
protect-interface interface-name;
```

### Hierarchy Level

```
[edit bridge-domains bridge-domain-name interface],
[edit logical-systems logical-systems--name bridge-domains bridge-domains interface],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name end-interface],
[edit logical-systems logical-systems-name routing-instances name interface ],
[edit protocols l2circuit local-switching interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name],
[edit protocols l2circuit local-switching interface interface-name end-interface]
[edit routing-instances routing-instances-name interface ]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Support at the following hierarchy levels introduced in Junos OS Release 17.4: **[edit bridge domains]**, **[edit logical-systems]**, and **[edit routing-instances]**.

### Description

Provide a backup for the protected interface in case of failure. Network traffic uses the primary interface only, as long as the primary interface functions.

### Options

***interface-name***—Name of the protect interface to configure.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring the Protect Interface | 73](#)

*Configuring EVPN Active-Standby Multihoming to a Single PE Device*

## protected-l2circuit

### Syntax

```
protected-l2circuit {
    egress-pe address;
    ingress-pe address;
    virtual-circuit-id identifier;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name
  egress-protection],
[edit protocols l2circuit neighbor address interface interface-name egress-protection]
```

### Release Information

Statement introduced in Junos OS release 10.4.

### Description

Configures the protected Layer 2 circuit as part of an egress protection virtual circuit (EPVC).

### Options

**egress-pe *address***—Specify the address of the egress PE router for the protected Layer 2 circuit.

**ingress-pe *address***—Specify the address of the ingress PE router for the protected Layer 2 circuit.

**virtual-circuit-id *identifier***—Specify the virtual circuit identifier for the protected Layer 2 circuit.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| *Example: Configuring an Egress Protection LSP for a Layer 2 Circuit*

## protector-interface

### Syntax

```
protector-interface interface-name;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name
  egress-protection],
[edit protocols l2circuit neighbor address interface interface-name egress-protection]
```

### Release Information

Statement introduced in Junos OS release 10.4.

### Description

Configures the protector interface for an egress protection LSP.

### Options

***interface-name***—Name of the interface used to protect traffic for an egress protection LSP.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| *Example: Configuring an Egress Protection LSP for a Layer 2 Circuit*

## protector-pe

### Syntax

```
protector-pe address {
  context-identifier identifier;
  lsp lsp-name;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name
  egress-protection],
[edit protocols l2circuit neighbor address interface interface-name egress-protection]
```

### Release Information

Statement introduced in Junos OS release 10.4.

### Description

Configures the protector PE router for an egress protection LSP. Test.

### Options

***address***—IPv4 address for the protector PE router.

**context-identifier *identifier***—Identifies the context for the egress protection LSP.

**lsp *lsp-name***—Specifies the LSP for the egress protection LSP.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

| *Example: Configuring an Egress Protection LSP for a Layer 2 Circuit*



## pseudowire-status-tlv

### Syntax

```
pseudowire-status-tlv hot-standby-vc-on;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],  
[edit protocols l2circuit neighbor address interface interface-name],
```

### Release Information

Statement introduced in Junos OS Release 10.0.

### Description

Enables the pseudowire type length variable (TLV). The pseudowire status TLV is used to communicate the status of a pseudowire back and forth between two PE routers. The pseudowire status TLV is configurable for each pseudowire connection and is disabled by default.

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

[Configuring the Pseudowire Status TLV | 74](#)

*Example: Configuring Pseudowire Redundancy in a Mobile Backhaul Scenario*

## psn-tunnel-endpoint

### Syntax

```
psn-tunnel-endpoint address;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name backup-neighbor
address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor address
backup-neighbor address],
[edit protocols l2circuit neighbor address interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name backup-neighbor address],
[edit routing-instances routing-instance-name protocols vpls neighbor address backup-neighbor address]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Hierarchy levels associated with the **backup-neighbor** statement added in Junos OS Release 9.2.

### Description

Specify the endpoint of the packet switched network (PSN) tunnel on the remote PE router.

### Options

***address***—Address for the tunnel endpoint.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring Layer 2 Circuits over Both RSVP and LDP LSPs | 75](#)

[Configuring Pseudowire Redundancy on the PE Router | 52](#)

## remote-site-id

### Syntax

```
remote-site-id remote-site-ID;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn site site-name
  interface interface-name],
[edit routing-instances routing-instance-name protocols l2vpn site site-name interface interface-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

### Description

Control the remote interface to which the interface should connect. If you do not explicitly configure the remote site ID, the order of the interfaces configured for the site determines the default value. This statement is optional.

### Options

***remote-site-ID***—Identifier specifying the interface on the remote PE router the Layer 2 VPN routing instance connects to.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring the Remote Site ID | 12](#)

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## routing-instances

### Syntax

```
routing-instances routing-instance-name { ... }
```

### Hierarchy Level

```
[edit],  
[edit logical-systems logical-system-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Configure an additional routing entity for a router or switch. You can create multiple instances of BGP, IS-IS, OSPF, OSPF version 3 (OSPFv3), and RIP for a router or switch.

### Default

Routing instances are disabled for the router or switch.

### Options

***routing-instance-name***—Name of the routing instance, a maximum of 31 characters. The remaining statements are explained separately.

***non-forwarding-vrf***—Enable this option to not create a routing and forwarding (VRF) table for local or transit routes belonging to the given VPN.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Configuring EVPN Routing Instances*

*Configuring Routing Instances on PE Routers in VPNs*

## send-oam

### Syntax

```
send-oam;
```

### Hierarchy Level

```
[edit protocols l2circuit neighbor address interface interface-name static]
```

### Release Information

Statement introduced in Junos OS Release 9.5.

Statement introduced in Junos OS Release 12.1x48 for PTX Series Packet Transport Routers.

Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro Routers.

### Description

Enable the ability to ping a static pseudowire. If you configure the **send-oam** statement, it applies to the backup neighbor as well. Once you have configured this statement, you can ping the static pseudowire by issuing the **ping mpls l2circuit** command.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring Static Layer 2 Circuits | 64](#)

[ping mpls l2circuit | 287](#)

## site (Layer 2 Circuits)

### Syntax

```
site site-name {
  hot-standby;
  site-identifier identifier;
  site-preference preference-value {
    backup;
    primary;
  }
  interface interface-name {
    description text;
    remote-site-id remote-site-ID;
  }
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn],
[edit routing-instances routing-instance-name protocols l2vpn]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

**hot-standby** option introduced in Junos OS Release 14.2 for MX Series routers.

### Description

Specify the site name, site identifier, and interfaces connecting to the site. Allows you to configure a remote site ID for remote sites.

### Options

**hot-standby**—Turn on the protector behavior for the site. This keeps backup pseudowire in continuous standby mode and ready for traffic forwarding.

**site-identifier** *identifier*—Numerical identifier for the site used as a default reference for the remote site ID.

**site-name**—Name of the site.

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

[Configuring the Site | 11](#)

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## site-identifier (Layer 2 Circuits)

### Syntax

```
site-identifier identifier;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn site site-name],
[edit routing-instances routing-instance-name protocols l2vpn site site-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

### Description

Specify the numerical identifier for the local Layer 2 VPN site.

### Options

***identifier***—The numerical identifier for the Layer 2 VPN site, which can be any number from 1 through 65,534.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring the Site | 11](#)

*Configuring an MPLS-Based Layer 2 VPN (CLI Procedure)*

## site-preference

### Syntax

```
site-preference preference-value {
    backup;
    primary;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn site site-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls site site-name],
[edit routing-instances routing-instance-name protocols l2vpn site site-name],
[edit routing-instances routing-instance-name protocols vpls site site-name]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced for Layer 2 VPNs in Junos OS Release 9.1.

### Description

Specify the preference value advertised for a particular Layer 2 VPN or VPLS site. The site preference value is encoded in the BGP local preference attribute. When a PE router receives multiple advertisements with the same VE identifier, the advertisement with the highest local preference value is preferred. You can use this statement to enable multihoming for Layer 2 VPNs and VPLS.

### Options

***preference-value***—Specify the preference value advertised for a Layer 2 VPN or VPLS site.

**Range:** 1 through 65,535

**backup**—Set the preference value to 1.

**primary**—Set the preference value to 65,535.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring a Site Preference and Layer 2 VPN Multihoming](#) | 14

[Configuring the VPLS Site Preference](#)



## source-attachment-identifier (Protocols VPWS)

### Syntax

```
source-attachment-identifier identifier;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn site site-name],  
[edit routing-instances routing-instance-name protocols l2vpn site site-name]
```

### Release Information

Statement introduced in Junos OS Release 13.2.

### Description

For FEC 129, specify the VPWS source attachment identifier. The point-to-point nature of VPWS requires that you specify the source access individual identifier (SAII) and the target access individual identifier (TAII). This SAII-TAII pair defines a unique pseudowire between two PE devices.

Auto-discovery routes are used by BGP to allow auto-discovery of remote source access individual identifiers (SAIIs) (the sources of the point-to-point pseudowires). One auto-discovery route is advertised for each source attachment identifier (SAI) in the instance or mesh group.

The SAII is specified with the **source-attachment-identifier** statement within the FEC 129 VPWS routing instance. You configure the source attachment identifier and the interfaces to associate with that source attachment identifier. Under each interface, you can configure the TAII with the **target-attachment-identifier** statement. If the configured target identifier matches a source identifier advertised by a remote PE device by way of a BGP auto-discovery message, the pseudowire between that source-target pair is signaled. If there is no match between an advertised source identifier and the configured target identifier, the pseudowire is not established.

### Options

**identifier**—The numerical identifier for the Layer 2 VPN site.

**Range:** 1 through 4,292,967,295

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Example: Configuring FEC 129 BGP Autodiscovery for VPWS*

| [target-attachment-identifier](#) | 262

## standby (Protocols Layer 2 Circuit)

### Syntax

```
standby;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name end-interface
  interface interface-name backup-neighbor address],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name backup-neighbor
  address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols vpls neighbor address
  backup-neighbor address],
[edit protocols l2circuit neighbor address interface interface-name backup-neighbor address],
[edit routing-instances routing-instance-name protocols vpls neighbor address backup-neighbor address]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Hierarchy levels associated with the **backup-neighbor** statement added in Junos OS Release 9.2.

### Description

Configure the pseudowire to the specified backup neighbor as the standby. When you configure this statement, traffic flows over both the active and standby pseudowires to the backup device (either a CE device or PE router). The backup device drops the traffic from the standby pseudowire, unless the active pseudowire fails. If the active pseudowire fails, the backup device automatically switches to the standby pseudowire.

The **standby** statement is quite similar to the **hot-standby** statement introduced in Junos OS Release 12.3. The **hot-standby** statement allows for a faster forwarding-path switchover during transition periods, as compared to what is allowed by the **standby** statement.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring Layer 2 Circuits over Both RSVP and LDP LSPs | 75](#)

[Configuring Pseudowire Redundancy on the PE Router | 52](#)

[Example: Configuring Layer 2 Circuit Switching Protection | 84](#)

## static (Protocols Layer 2 Circuit)

### Syntax

```
static {
    incoming-label label;
    outgoing-label label;
    send-oam;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name l2circuit neighbor address interface interface-name backup-neighbor
  neighbor],
[edit protocols l2circuit neighbor address interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name backup-neighbor neighbor]
```

### Release Information

Statement introduced in Junos OS Release 9.5.

Statement introduced in Junos OS Release 12.1X48 for PTX Series routers.

Statement introduced in Junos OS Release 12.2 for ACX Series routers.

### Description

Configures static Layer 2 circuit pseudowires. Static pseudowires are designed for networks that do not support LDP or do not have LDP enabled. You configure a static pseudowire by configuring static values for the in and out labels needed to enable a pseudowire connection.

### Options

**incoming-label**—(Optional for PTX Series Packet Transport Routers only) Configure the Layer 2 circuit incoming static pseudowire label.

**Range:** 1000000 through 1048575

**outgoing-label**—(Optional for PTX Series Packet Transport Routers only) Configure the Layer 2 circuit outgoing static pseudowire label.

**Range:** 16 through 1048575

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

| [Configuring Static Layer 2 Circuits](#) | 64

## target-attachment-identifier (Protocols VPWS)

### Syntax

```
target-attachment-identifier identifier;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances instance-name protocols l2vpn site site-name interface interface-name],  
[edit routing-instances instance-name protocols l2vpn site site-name interface interface-name]
```

### Release Information

Statement introduced in Junos OS Release 13.2.

### Description

For FEC 129, specify the VPWS target attachment identifier. The point-to-point nature of VPWS requires that you specify the source access individual identifier (SAII) and the target access individual identifier (TAII). This SAII-TAII pair defines a unique pseudowire between two PE devices.

Auto-discovery routes are used by BGP to allow auto-discovery of SAIIs (the sources of the point-to-point pseudowires). One auto-discovery route is advertised for each source attachment identifier (SAI) in the instance or mesh group.

The SAII is specified with the **source-attachment-identifier** statement within the FEC 129 VPWS routing instance. You configure the source attachment identifier and the interfaces to associate with that source attachment identifier. Under each interface, you can configure the TAI with the **target-attachment-identifier** statement. If the configured target identifier matches a source identifier advertised by a remote PE device by way of a BGP auto-discovery message, the pseudowire between that source-target pair is signaled. If there is no match between an advertised source identifier and the configured target identifier, the pseudowire is not established.

### Options

**identifier**—The numerical identifier for the Layer 2 VPN site.

**Range:** 1 through 4,292,967,295

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

Example: Configuring FEC 129 BGP Autodiscovery for VPWS

[source-attachment-identifier](#) | [257](#)

## template

### Syntax

```
template;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols mpls label-switched-path p2mp-lsp-template-name],  
[edit protocols mpls label-switched-path p2mp-lsp-template-name]
```

### Release Information

Statement introduced in Junos OS Release 8.3.

Statement introduced in Junos OS Release 12.3X50 for the QFX Series.

### Description

Specify a template for the dynamically generated point-to-multipoint LSPs used for VPLS flooding.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Configuring Dynamic Point-to-Multipoint Flooding LSPs with a Preconfigured Template*

## traceoptions (Egress Protection)

### Syntax

```
traceoptions {
  file filename <files number> <size size> <world-readable | no-world-readable>;
  flag flag;
}
```

### Hierarchy Level

```
[edit protocols mpls egress-protection],
```

### Release Information

Statement introduced in Junos OS Release 11.4R3.

### Description

Configure tracing operations for egress protection.

### Options

**filename**—Name of the file to receive the output of the tracing operation.

**files *number***—(Optional) Maximum number of trace files. If you specify a maximum number of files, you must also include the **size** statement to specify the maximum file size. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file.0*, then *trace-file.1*, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

**Range:** 2 through 1000

**Default:** 2 files

**flag**—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements.

- **all**—Trace all operations
- **error**—Trace error conditions
- **route**—Trace route transitions
- **state**—Trace state transitions

### Required Privilege Level

routing—To view this statement in the configuration.

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



## RELATED DOCUMENTATION

| *Example: Configuring MPLS Egress Protection for Layer 3 VPN Services*

## traceoptions (Protocols Layer 2 Circuit)

### Syntax

```
traceoptions {
  file filename <files number> <size size> <world-readable | no-world-readable>;
  flag flag <flag-modifier> <disable>;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit],
[edit protocols l2circuit]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Trace traffic flowing through a Layer 2 circuit.

### Options

**disable**—(Optional) Disable the tracing operation. You can use this option to disable a single operation when you have defined a broad group of tracing operations, such as **all**.

**file *filename***—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks (" ").

**files *number***—(Optional) Maximum number of trace files. When a trace file named **trace-file** reaches its maximum size, it is renamed **trace-file.0**, then **trace-file.1**, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

If you specify a maximum number of files, you also must specify a maximum file size with the **size** option.

**Range:** 2 through 1000 files

**Default:** 2 files

**flag *flag***—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements.

- **connections**—Layer 2 circuit connections (events and state changes)
- **error**—Error conditions
- **fec**—Layer 2 circuit advertisements received or sent by means of LDP
- **topology**—Layer 2 circuit topology changes caused by reconfiguration or advertisements received from other PE routers

**flag-modifier**—(Optional) Modifier for the tracing flag. You can specify the **detail** modifier if you want to provide detailed trace information.

**no-world-readable**—(Optional) Do not allow any user to read the log file.

**size size**—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file.0*. When the *trace-file* again reaches its maximum size, *trace-file.0* is renamed *trace-file.1* and *trace-file* is renamed *trace-file.0*. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

If you specify a maximum file size, you also must specify a maximum number of trace files with the **files** option.

**Syntax:** **xk** to specify kilobytes, **xm** to specify megabytes, or **xg** to specify gigabytes

**Range:** 10 KB through the maximum file size supported on your system

**Default:** 1 MB

**world-readable**—(Optional) Allow any user to read the log file.

#### Required Privilege Level

routing—To view this statement in the configuration.

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

#### RELATED DOCUMENTATION

| [Tracing Layer 2 Circuit Operations](#) | 116

## traceoptions (Protocols Layer 2 VPN)

### Syntax

```
traceoptions {
  file filename <files number> <size size> <world-readable | no-world-readable>;
  flag flag <flag-modifier> <disable>;
}
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols l2vpn],
[edit routing-instances routing-instance-name protocols l2vpn]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Trace traffic flowing through a Layer 2 VPN.

### Options

**disable**—(Optional) Disable the tracing operation. You can use this option to disable a single operation when you have defined a broad group of tracing operations, such as **all**.

**file *filename***—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks (" ").

**files *number***—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file.0*, then *trace-file.1*, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

If you specify a maximum number of files, you also must specify a maximum file size with the **size** option.

**Range:** 2 through 1000 files

**Default:** 2 files

**flag *flag***—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements.

- **all**—All Layer 2 VPN tracing options
- **connections**—Layer 2 connections (events and state changes)
- **error**—Error conditions
- **general**—General events

- **nlri**—Layer 2 advertisements received or sent by means of the BGP
- **normal**—Normal events
- **policy**—Policy processing
- **route**—Routing information
- **state**—State transitions
- **task**—Routing protocol task processing
- **timer**—Routing protocol timer processing
- **topology**—Layer 2 VPN topology changes caused by reconfiguration or advertisements received from other PE routers using BGP

**flag-modifier**—(Optional) Modifier for the tracing flag. You can specify the following modifier:

- **detail**—Provide detailed trace information
- **receive**—Trace received packets
- **send**—Trace transmitted packets

**no-world-readable**—(Optional) Prevents any user from reading the trace file.

**size size**—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file.0*. When *trace-file* again reaches its maximum size, *trace-file.0* is renamed *trace-file.1* and *trace-file* is renamed *trace-file.0*. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

If you specify a maximum file size, you also must specify a maximum number of trace files with the **files** option.

**Syntax:** **xk** to specify kilobytes, **xm** to specify megabytes, or **xg** to specify gigabytes

**Range:** 10 KB through the maximum file size supported on your system

**Default:** 1 MB

**world-readable**—(Optional) Allow any user to read the trace file.

**Default:** The default is **no-world-readable**.

#### Required Privilege Level

routing—To view this statement in the configuration.

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

#### RELATED DOCUMENTATION

## transmit-interval (BFD Liveness Detection)

### Syntax

```
transmit-interval {
    minimum-interval milliseconds;
    threshold milliseconds;
}
```

### BGP

```
[edit logical-systems name protocols bgp bfd-liveness-detection],
[edit logical-systems name protocols bgpgroup bfd-liveness-detection],
[edit logical-systems name protocols bgp group name neighbor address bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols bgp bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols bgpgroup bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols bgpgroup neighbor address bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols bgp bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols bgpgroup bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols bgpgroup neighbor address
    bfd-liveness-detection],
[edit protocols bgp bfd-liveness-detection],
[edit protocols bgp group bfd-liveness-detection],
[edit protocols bgp group neighbor address bfd-liveness-detection],
[edit routing-instances name protocols bgp bfd-liveness-detection],
[edit routing-instances name protocols bgp group bfd-liveness-detection],
[edit routing-instances name protocols bgp group neighbor address bfd-liveness-detection],
[edit tenants name routing-instances name protocols bgp bfd-liveness-detection]
[edit tenants name routing-instances name protocols bgp group bfd-liveness-detection]
[edit tenants name routing-instances name protocols bgp groupneighbor address bfd-liveness-detection]
```

### EVPN, L2VPN, VPLS

```
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
    bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name
    neighbor neighbor-id oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam
    bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)neighbor neighbor-id
    oam bfd-liveness-detection],
```

```
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group
  mesh-group-name neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor neighbor-id oam
  bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
  bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor
  neighbor-id oam bfd-liveness-detection],
```

### Release Information

Statement introduced in Junos OS Release 8.2.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Support for BFD authentication introduced in Junos OS Release 9.6.

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPN and VPLS.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Specify the transmit interval for the **bfd-liveness-detection** statement. The negotiated transmit interval for a peer is the interval between the sending of BFD packets to peers. The receive interval for a peer is the minimum time that it requires between packets sent from its peer; the receive interval is not negotiated between peers. To determine the transmit interval, each peer compares its configured minimum transmit interval with its peer's minimum receive interval. The larger of the two numbers is accepted as the transmit interval for that peer.

## Options

**minimum-interval *milliseconds***— Configure the minimum interval at which the local routing device transmits hello packets to a neighbor with which it has established a BFD session. Optionally, instead of using this statement at this hierarchy level, you can configure the minimum transmit interval using the **minimum-interval** statement at the **bfd-liveness-detection** hierarchy level.

**NOTE:** The threshold value specified in the **threshold** statement must be greater than the value specified in the **minimum-interval** statement for the **transmit-interval** statement.

**Range:** 1 through 255,000 milliseconds

**threshold *milliseconds***— Specify the threshold for the adaptation of the BFD session transmit interval. When the transmit interval adapts to a value greater than the threshold, a single trap and a single system message are sent.

**Range:** 0 through 4,294,967,295 ( $2^{32} - 1$ )

**NOTE:** The threshold value specified in the **threshold** statement must be greater than the value specified in the **minimum-interval** statement for the **transmit-interval** statement.

## Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring BFD for Layer 2 VPN and VPLS | 56](#)

*Example: Configuring BFD for Static Routes for Faster Network Failure Detection*

*bfd-liveness-detection (BGP)*

[bfd-liveness-detection \(Layer 2 VPN and VPLS\) | 157](#)



## version (BFD Liveness Detection)

### Syntax

```
version (1 | automatic);
```

### Hierarchy Level

```
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit logical-systems name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name
neighbor neighbor-id oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam
bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)neighbor neighbor-id
oam bfd-liveness-detection],
[edit logical-systems name tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group
mesh-group-name neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam bfd-liveness-detection],
[edit routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) oam bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls) neighbor neighbor-id oam
bfd-liveness-detection],
[edit tenants name routing-instances name protocols (evpn | l2vpn | vpls)mesh-group mesh-group-name neighbor
neighbor-id oam bfd-liveness-detection],
```

### Release Information

Statement introduced in Junos OS Release 8.1

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Statement introduced in Junos OS Release 13.2 for Layer 2 VPN and VPLS.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Specify the BFD version for detection. You can explicitly configure BFD version 1 or the routing device can automatically detect the BFD version. By default, the routing device automatically detects the BFD version.

### Options

Configure the BFD version to detect: **1** (BFD version 1) or **automatic** (autodetect the BFD version).

**Default:** The routing device automatically detects the BFD version.

**Required Privilege Level**

routing—To view this statement in the configuration.

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

**RELATED DOCUMENTATION**

---

[Configuring BFD for Layer 2 VPN and VPLS | 56](#)

---

*Example: Configuring BFD Authentication for BGP*

---

*Example: Configuring BFD on Internal BGP Peer Sessions*

---

*Understanding BFD Authentication for BGP*

## virtual-circuit-id

### Syntax

```
virtual-circuit-id identifier;
```

### Hierarchy Level

```
[edit logical-systems logical-system-name protocols l2circuit local-switching interface interface-name backup-neighbor
address],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name],
[edit logical-systems logical-system-name protocols l2circuit neighbor address interface interface-name backup-neighbor
address],
[edit protocols l2circuit local-switching interface interface-name backup-neighbor address],
[edit protocols l2circuit neighbor address interface interface-name],
[edit protocols l2circuit neighbor address interface interface-name backup-neighbor address]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Hierarchy levels for **backup-neighbor** (pseudowire redundancy) added in Junos OS Release 9.2.

Statement introduced in Junos OS Release 11.1 for EX Series switches.

Statement introduced in Junos OS Release 12.3 at the **[edit protocols l2circuit local-switching interface *interface-name* backup-neighbor *address*]** hierarchy level.

### Description

Uniquely identify a Layer 2 circuit for either a standard pseudowire or a redundant pseudowire.

### Options

***identifier***—1 through 4,294,967,295

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

[Configuring the Virtual Circuit ID | 76](#)

[Configuring Pseudowire Redundancy on the PE Router | 52](#)

[Example: Configuring Layer 2 Circuit Switching Protection | 84](#)

## vlan-id

### Syntax

```
vlan-id number;
```

### 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 before Junos OS Release 7.4.

### Description

For Fast Ethernet and Gigabit Ethernet interfaces only, bind an 802.1Q VLAN tag ID to a logical interface.

### Options

***number***—A valid VLAN identifier.

**Range:** For 4-port Fast Ethernet PICs configured to handle VPLS traffic, 512 through 1023. For 1-port and 10-port Gigabit Ethernet PICs configured to handle VPLS traffic, 512 through 4094.

**NOTE:** On Junos OS Evolved, **vlan-id 0** is not supported.

### Required Privilege Level

interface—To view this statement in the configuration.

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

### RELATED DOCUMENTATION

| *Configuring Interfaces for VPLS Routing*

## vlan-id (routing instance)

### Syntax

```
vlan-id (vlan-id | all | none);
```

### Hierarchy Level

```
[edit logical-systems logical-system-name routing-instances routing-instance-name],  
[edit routing-instances routing-instance-name]  
[edit routing-instances routing-instance-name instance-type]
```

### Release Information

Statement introduced in Junos OS Release 13.2.

Statement introduced in Junos OS Release 14.2 for EX Series switches.

Statement introduced in Junos OS Release 17.1 for the QFX Series.

### Description

Specify 802.1Q VLAN tag IDs to a routing instance.

### Options

**vlan-id**—A valid VLAN identifier.

**Range:** For 4-port Fast Ethernet PICs, 512 through 1023. For 1-port and 10-port Gigabit Ethernet PICs configured to handle VPLS traffic, 512 through 4094.

**all**—Include all VLAN identifiers specified on the logical interfaces included in the routing instance.

**none**—Include no VLAN identifiers for the routing instance.

### Required Privilege Level

routing—To view this statement in the configuration.

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

## RELATED DOCUMENTATION

*Configuring EVPN Routing Instances*

*Configuring EVPN Routing Instances on EX9200 Switches*

## vlan-tagging

### Syntax

```
vlan-tagging;
```

### Syntax (QFX Series, NFX Series, and EX4600)

```
vlan-tagging;
```

### Syntax (SRX Series Interfaces)

```
vlan-tagging native-vlan-id vlan-id;
```

### Hierarchy Level

```
[edit interfaces interface-name],  
[edit logical-systems logical-system-name interfaces interface-name]
```

### QFX Series, NFX Series, and EX4600 Interfaces

```
[edit interfaces (QFX Series) interface-name ]  
[edit interfaces (QFX Series) interface-range interface-range-name ]
```

### SRX Series Interfaces

```
[edit interfaces interface ]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

Statement introduced in Junos OS Release 9.0 for EX Series switches.

Statement introduced in Junos OS Release 9.5.

Statement introduced in Junos OS Release 11.3 for the QFX Series.

Statement introduced in Junos OS Release 12.2 for ACX Series Universal Metro Routers.

Statement introduced in Junos OS Release 13.2 for PTX Series Routers.

Statement introduced in Junos OS Release 14.1X53-D10 for the QFX Series.

### Description

For Fast Ethernet and Gigabit Ethernet interfaces, aggregated Ethernet interfaces configured for VPLS, and pseudowire subscriber interfaces, enable the reception and transmission of 802.1Q VLAN-tagged frames on the interface.

**NOTE:** For QFX Series configure VLAN identifier for untagged packets received on the physical interface of a trunk mode interface. Enable VLAN tagging. The platform receives and forwards single-tag frames with 802.1Q VLAN tags.

On EX Series switches except for EX4300 and EX9200 switches, the **vlan-tagging** and **family ethernet-switching** statements cannot be configured on the same interface. Interfaces on EX2200, EX3200, EX3300, EX4200, and EX4500 switches are set to **family ethernet-switching** by the default factory configuration. EX6200 and EX8200 switch interfaces do not have a default **family** setting.

### Default

VLAN tagging is disabled by default.

### Options

**native-vlan-id**— (SRX Series) 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

[802.1Q VLANs Overview](#)

[Configuring a Layer 3 Subinterface \(CLI Procedure\)](#)

[Configuring Tagged Aggregated Ethernet Interfaces](#)

[Example: Configuring Layer 3 Subinterfaces for a Distribution Switch and an Access Switch](#)

[vlan-id](#)

[Configuring a Layer 3 Logical Interface](#)

[Configuring VLAN Tagging](#)



# Operational Commands

## IN THIS CHAPTER

- clear pim snooping join | 282
- clear pim snooping statistics | 284
- ping mpls l2circuit | 287
- ping mpls l2vpn | 290
- request l2circuit-switchover | 293
- show interfaces lsi (Label-Switched Interface) | 295
- show l2circuit connections | 299
- show l2vpn connections | 310
- show pim snooping interfaces | 319
- show pim snooping join | 323
- show pim snooping neighbors | 328
- show pim snooping statistics | 335
- show route | 341

## clear pim snooping join

### Syntax

```
clear pim snooping join  
<instance instance-name>  
<logical-system logical-system-name>  
<vlan-id vlan-id>
```

### Release Information

Command introduced in Junos OS Release 12.3 for MX Series 5G Universal Routing Platforms.

Command introduced in Junos OS Release 13.2 for M Series Multiservice Edge devices.

### Description

Clear information about Protocol Independent Multicast (PIM) snooping joins.

### Options

**none**—Display detailed information.

**instance *instance-name***—(Optional) Clear PIM snooping join information for the specified routing instance.

**logical-system *logical-system-name***—(Optional) Delete the IGMP snooping statistics for a given logical system or for all logical systems.

**vlan-id *vlan-identifier***—(Optional) Clear PIM snooping join information for the specified VLAN.

### Required Privilege Level

view

### RELATED DOCUMENTATION

| [PIM Snooping for VPLS](#)

### List of Sample Output

[clear pim snooping join on page 283](#)

### Output Fields

See [show pim snooping join](#) for an explanation of the output fields.

## Sample Output

### clear pim snooping join

The following sample output displays information about PIM snooping joins before and after the **clear pim snooping join** command is entered:

```
user@host> show pim snooping join extensive
```

```
Instance: vpls1
Learning-Domain: vlan-id 10
Learning-Domain: vlan-id 20

Group: 198.51.100.2
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 192.0.2.5, port: ge-1/3/7.20
Downstream port: ge-1/3/1.20
Downstream neighbors:
192.0.2.2 State: Join Flags: SRW Timeout: 185

Group: 198.51.100.3
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 192.0.2.4, port: ge-1/3/5.20
Downstream port: ge-1/3/3.20
Downstream neighbors:
192.0.2.3 State: Join Flags: SRW Timeout: 175
```

```
user@host> clear pim snooping join
```

```
Clearing the Join/Prune state for 203.0.113.0/24
Clearing the Join/Prune state for 203.0.113.0/24
```

```
user@host> show pim snooping join extensive
```

```
Instance: vpls1
Learning-Domain: vlan-id 10
Learning-Domain: vlan-id 20
```

## clear pim snooping statistics

### Syntax

```
clear pim snooping statistics
<instance instance-name>
<interface interface-name>
<logical-system logical-system-name>
<vlan-id vlan-id>
```

### Release Information

Command introduced in Junos OS Release 12.3 for MX Series 5G Universal Routing Platforms.

Command introduced in Junos OS Release 13.2 for M Series Multiservice Edge devices.

### Description

Clear Protocol Independent Multicast (PIM) snooping statistics.

### Options

**none**—Clear PIM snooping statistics for all family addresses, instances, and interfaces.

**instance *instance-name***—(Optional) Clear statistics for a specific PIM-snooping-enabled routing instance.

**interface *interface-name***—(Optional) Clear PIM snooping statistics for a specific interface.

**logical-system *logical-system-name***—(Optional) Delete the IGMP snooping statistics for a given logical system or for all logical systems.

**vlan-id *vlan-identifier***—(Optional) Clear PIM snooping statistics information for the specified VLAN.

### Required Privilege Level

clear

### RELATED DOCUMENTATION

| [PIM Snooping for VPLS](#)

### List of Sample Output

[clear pim snooping statistics on page 285](#)

### Output Fields

See [show pim snooping statistics](#) for an explanation of the output fields.

## Sample Output

### clear pim snooping statistics

The following sample output displays PIM snooping statistics before and after the **clear pim snooping statistics** command is entered:

```
user@host> show pim snooping statistics
```

```
Instance: vpls1
Learning-Domain: vlan-id 10

Tx J/P messages 0
RX J/P messages 660
Rx J/P messages -- seen 0
Rx J/P messages -- received 660
Rx Hello messages 1396
Rx Version Unknown 0
Rx Neighbor Unknown 0
Rx Upstream Neighbor Unknown 0
Rx Bad Length 0
Rx J/P Busy Drop 0
Rx J/P Group Aggregate 0
Rx Malformed Packet 0

Learning-Domain: vlan-id 20
```

```
user@host> clear pim snooping statistics
```

```
user@host> show pim snooping statistics
```

```
Instance: vpls1
Learning-Domain: vlan-id 10

Tx J/P messages 0
RX J/P messages 0
Rx J/P messages -- seen 0
Rx J/P messages -- received 0
Rx Hello messages 0
Rx Version Unknown 0
Rx Neighbor Unknown 0
Rx Upstream Neighbor Unknown 0
Rx Bad Length 0
Rx J/P Busy Drop 0
Rx J/P Group Aggregate 0
```

Rx Malformed Packet 0

Learning-Domain: vlan-id 20

## ping mpls l2circuit

### Syntax

```
ping mpls l2circuit (interface interface-name | virtual-circuit virtual-circuit-id neighbor address)
<count count>
<destination address>
<detail>
<exp forwarding-class>
<logical-system (all | logical-system-name)>
reply-mode (application-level-control-channel | ip-udp | no-reply)
<size bytes>
<source source-address>
<sweep>
<v1>
```

### Release Information

Command introduced before Junos OS Release 7.4.

Command introduced in Junos OS Release 9.0 for EX Series switches.

Statement introduced in Junos OS Release 12.3X50 for the QFX Series.

The **size** and **sweep** options were introduced in Junos OS Release 9.6.

The **reply-mode** option and its suboptions are introduced in Junos OS Release 10.4R1.

### Description

Check the operability of the MPLS Layer 2 circuit connections. Type Ctrl+c to interrupt a **ping mpls l2circuit** command.

**NOTE:** This command is not supported on EX4500 and EX4550 switches.

### Options

**count** *count*—(Optional) Number of ping requests to send. If **count** is not specified, five ping requests are sent. The range of values is **1** through **1,000,000**. The default value is **5**.

**destination** *address*—(Optional) Specify an address other than the default (**127.0.0.1/32**) for the ping echo requests. The address can be anything within the **127/8** subnet.

**detail**—(Optional) Display detailed information about the echo requests sent and received.

**exp** *forwarding-class*—(Optional) Value of the forwarding class for the MPLS ping packets.

**interface** *interface-name*—Ping an interface configured for the Layer 2 circuit on the egress provider edge (PE) router.

**logical-system** (**all** | *logical-system-name*)—(Optional) Perform this operation on all logical systems or on the specified logical system.

**reply-mode**—(Optional) Reply mode for the ping request. This option has the following suboptions:

**application-level-control-channel**—Reply using an application level control channel.

**ip-udp**—Reply using an IPv4 or IPv6 UDP packet.

**no-reply**—Do not reply to the ping request.

**NOTE:** The **reply-mode** option and its suboptions **application-level-control-channel**, **ip-udp**, and **no-reply** are also available in Junos OS Release 10.2R4 and 10.3R2.

**size bytes**—(Optional) Size of the label-switched path (LSP) ping request packet (**96** through **65468** bytes). Packets are 4-byte aligned. For example, If you enter a size of 97, 98, 99, or 100, the router or switch uses a size value of 100 bytes. If you enter a packet size that is smaller than the minimum size, an error message is displayed reminding you of the 96-byte minimum.

**source source-address**—(Optional) IP address of the outgoing interface. This address is sent in the IP source address field of the ping request. If this option is not specified, the default address is usually the loopback interface (**lo.0**).

**sweep**—(Optional) Automatically determine the size of the maximum transmission unit (MTU).

**v1**—(Optional) Use the type 9 Layer 2 circuit type, length, and value (TLV).

**virtual-circuit virtual-circuit-id neighbor address**—Ping the virtual circuit identifier on the egress PE router or switch and the specified neighbor, testing the integrity of the Layer 2 circuit between the ingress and egress PE routers or switches.

### Additional Information

You must configure MPLS at the **[edit protocols mpls]** hierarchy level on the egress PE router or switch (the router or switch receiving the MPLS echo packets) to ping a Layer 2 circuit.

In asymmetric MTU scenarios, the echo response might be dropped. For example, if the MTU from System A to System B is 1000 bytes, the MTU from System B to System A is 500 bytes, and the ping request packet size is 1000 bytes, the echo response is dropped because the PAD TLV is included in the echo response, making it too large.

### Required Privilege Level

network

### List of Sample Output

[ping mpls l2circuit interface on page 289](#)



[ping mpls l2circuit virtual-circuit detail on page 289](#)

[ping mpls l2circuit interface <interface-name> reply-mode on page 289](#)

### Output Fields

When you enter this command, you are provided feedback on the status of your request. An exclamation point (!) indicates that an echo reply was received. A period (.) indicates that an echo reply was not received within the timeout period. An x indicates that an echo reply was received with an error code. Packets with an error code are not counted in the received packets count. They are accounted for separately.

## Sample Output

### ping mpls l2circuit interface

```
user@host> ping mpls l2circuit interface so-1/0/0.1
```

```
Request for seq 1, to interface 69, labels <100000, 100208>, packet size 100
Reply for seq 1, return code: Egress-ok, time: 0.439 ms
```

### ping mpls l2circuit virtual-circuit detail

```
user@host> ping mpls l2circuit virtual-circuit 200 neighbor 10.255.245.122/32 detail
```

```
Request for seq 1, to interface 68, labels <100048, 100128>, packet size 100

Reply for seq 1, return code: Egress-ok time: 0.539 ms
```

### ping mpls l2circuit interface <interface-name> reply-mode

```
user@host> ping mpls l2circuit interface lt-1/2/0.21 reply-mode application-level-control-channel
```

```
!!!!
--- lsping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

## ping mpls l2vpn

### Syntax

```
ping mpls l2vpn (instance instance-name local-site-id local-site-id-number remote-site-id remote-site-id-number |
  interface interface-name)
<bottom-label-ttl>
<count count>
<destination address>
<detail>
<exp forwarding-class>
<logical-system (all | logical-system-name)>
reply-mode (application-level-control-channel | ip-udp | no-reply)
<size bytes>
<source source-address>
<sweep>
```

### Release Information

Command introduced before Junos OS Release 7.4.

Command introduced in Junos OS Release 9.0 for EX Series switches.

The **size** and **sweep** options were introduced in Junos OS Release 9.6.

The **reply-mode** option and its suboptions are introduced in Junos OS Release 10.4R1.

### Description

Check the operability of MPLS Layer 2 virtual private network (VPN) connections. Type Ctrl+c to interrupt a **ping mpls l2vpn** command.

### Options

**bottom-label-ttl**—(Optional) Display the time-to-live value for the bottom label in the label stack.

**count *count***—(Optional) Number of ping requests to send. If **count** is not specified, five ping requests are sent. The range of values is 1 through 1,000,000. The default value is 5.

**destination *address***—(Optional) Specify an address other than the default (**127.0.0.1/32**) for the ping echo requests. The address can be anything within the **127/8** subnet.

**detail**—(Optional) Display detailed information about the echo requests sent and received.

**exp *forwarding-class***—(Optional) Value of the forwarding class for the MPLS ping packets.

**instance *instance-name* local-site-id *local-site-id-number* remote-site-id *remote-site-id-number***—Ping a combination of the Layer 2 VPN routing instance name, the local site identifier, and the remote site identifier, testing the integrity of the Layer 2 VPN circuit (specified by the identifiers) between the ingress and egress provider edge (PE) routers or switches.

**interface *interface-name***—Ping an interface configured for the Layer 2 VPN on the egress PE router or switch.

**logical-system (all | *logical-system-name*)**—(Optional) Perform this operation on all logical systems or on the specified logical system.

**reply-mode**—(Optional) Reply mode for the ping request. This option has the following suboptions:

**application-level-control-channel**—Reply using an application level control channel.

**ip-udp**—Reply using an IPv4 or IPv6 UDP packet.

**no-reply**—Do not reply to the ping request.

The **reply-mode** option and its suboptions **application-level-control-channel**, **ip-udp**, and **no-reply** are also available in Junos OS Release 10.2R4 and 10.3R2.

**size *bytes***—(Optional) Size of the label-switched path (LSP) ping request packet (**96** through **65468** bytes). Packets are 4-byte aligned. For example, If you enter a size of 97, 98, 99, or 100, the router or switch uses a size value of 100 bytes. If you enter a packet size that is smaller than the minimum size, an error message is displayed reminding you of the 96-byte minimum.

**source *source-address***—(Optional) IP address of the outgoing interface. This address is sent in the IP source address field of the ping request. If this option is not specified, the default address is usually the loopback interface (**lo.0**).

**sweep**—(Optional) Automatically determine the size of the maximum transmission unit (MTU).

### Additional Information

You must configure MPLS at the **[edit protocols mpls]** hierarchy level on the egress PE router or switch (the router or switch receiving the MPLS echo packets) to ping a Layer 2 circuit.

In asymmetric MTU scenarios, the echo response might be dropped. For example, if the MTU from System A to System B is 1000 bytes, the MTU from System B to System A is 500 bytes, and the ping request packet size is 1000 bytes, the echo response is dropped because the PAD TLV is included in the echo response, making it too large.

### Required Privilege Level

network

### List of Sample Output

[ping mpls l2vpn instance on page 292](#)

[ping mpls l2vpn instance detail on page 292](#)

[ping mpls l2vpn interface <interface-name> reply-mode on page 292](#)

### Output Fields

When you enter this command, you are provided feedback on the status of your request. An exclamation point (!) indicates that an echo reply was received. A period (.) indicates that an echo reply was not received within the timeout period. An x indicates that an echo reply was received with an error code these packets are not counted in the received packets count. They are accounted for separately.

## Sample Output

### ping mpls l2vpn instance

```
user@host> ping mpls l2vpn instance vpn1 remote-site-id 1 local-site-id 2
```

```
!!!!
--- lsping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

### ping mpls l2vpn instance detail

```
user@host> ping mpls l2vpn instance vpn1 remote-site-id 1 local-site-id 2 detail
```

```
Request for seq 1, to interface 68, labels <800001, 100176>
Reply for seq 1, return code: Egress-ok
Request for seq 2, to interface 68, labels <800001, 100176>
Reply for seq 2, return code: Egress-ok
Request for seq 3, to interface 68, labels <800001, 100176>
Reply for seq 3, return code: Egress-ok
Request for seq 4, to interface 68, labels <800001, 100176>
Reply for seq 4, return code: Egress-ok
Request for seq 5, to interface 68, labels <800001, 100176>
Reply for seq 5, return code: Egress-ok

--- lsping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

### ping mpls l2vpn interface <interface-name> reply-mode

```
user@host> ping mpls l2vpn interface lt-1/2/0.21 reply-mode ip-udp
```

```
!!!!
--- lsping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

## request l2circuit-switchover

### Syntax

```
request l2circuit-switchover
<logical-system (all | logical-system-name)>
<neighbor address>
<virtual-circuit-id identifier>
```

### Release Information

Command introduced in Junos OS Release 9.2.

Statement introduced in Junos OS Release 14.1X53-D10 for the QFX Series and for EX4600 switches.

### Description

Manually trigger a switch from the active pseudowire to the redundant pseudowire. This command can be useful when performing network maintenance.

### Options

**logical-system (all | *logical-system-name*)**—(Optional) Perform this operation on all logical systems or on a particular logical system.

**neighbor *address***—(Optional) Trigger a switch of all of the active pseudowire connections with the specified neighbor to their respective redundant pseudowires.

**virtual-circuit-id *identifier***—(Optional) Trigger a switch from the active pseudowire connection of the specified Layer 2 circuit to its redundant pseudowire.

### Required Privilege Level

maintenance

## RELATED DOCUMENTATION

| *MPLS Feature Support on QFX Series and EX4600 Switches*

### List of Sample Output

[request l2circuit-switchover virtual-circuit-id on page 294](#)

### Output Fields

When you enter this command, you are provided feedback on the status of your request.

## Sample Output

```
request l2circuit-switchover virtual-circuit-id
```

```
user@host>request l2circuit-switchover virtual-circuit-id 12
```

## show interfaces lsi (Label-Switched Interface)

### Syntax

```
show interfaces interface-type
<brief | detail | extensive | terse>
<descriptions>
<media>
<routing-instance instance-name>
<snmp-index snmp-index>
<statistics>
```

### Release Information

Command introduced before Junos OS Release 7.4.

### Description

Display status information about the specified label-switched interface (LSI).

### Options

***interface-type***—On most routers, the interface type is **lt-fpc/pic/port**.

**brief | detail | extensive | terse**—(Optional) Display the specified level of output.

**descriptions**—(Optional) Display interface description strings.

**media**—(Optional) Display media-specific information about network interfaces.

**routing-instance *instance-name***—(Optional) Display information for the specified routing instance.

**snmp-index *snmp-index***—(Optional) Display information for the specified SNMP index of the interface.

**statistics**—(Optional) Display static interface statistics.

### Required Privilege Level

view

## RELATED DOCUMENTATION

### List of Sample Output

[show interfaces lsi extensive on page 297](#)

### Output Fields

[Table 7 on page 296](#) lists the output fields for the **show interfaces** (logical tunnel) command. Output fields are listed in the approximate order in which they appear.

Table 7: Logical Tunnel show interfaces Output Fields

Field Name	Field Description	Level of Output
<b>Physical Interface</b>		
Physical interface	Name of the physical interface.	All levels
<b>Logical Interface</b>		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	<b>detail extensive none</b>
SNMP ifIndex	SNMP interface index number.	<b>detail extensive none</b>
Generation	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
Flags	Information about the logical interface. Possible values are described in the “Logical Interface Flags” section under <i>Common Output Fields Description</i> .	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
Traffic statistics	<p>Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.</p> <ul style="list-style-type: none"> <li>• <b>Input bytes</b>—Rate of bytes received on the interface.</li> <li>• <b>Output bytes</b>—Rate of bytes transmitted on the interface.</li> <li>• <b>Input packets</b>—Rate of packets received on the interface.</li> <li>• <b>Output packets</b>—Rate of packets transmitted on the interface.</li> </ul>	<b>detail extensive</b>
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	<b>detail extensive</b>
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	<b>detail extensive</b>



Table 7: Logical Tunnel show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
Protocol	Protocol family configured on the logical interface, such as <b>iso</b> , <b>inet6</b> , <b>mpls</b> .	<b>detail extensive none</b>
MTU	MTU size on the logical interface.	<b>detail extensive none</b>
Generation	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
Flags	Information about the protocol family flags. Possible values are described in the “Family Flags” section under <i>Common Output Fields Description</i> .	<b>detail extensive none</b>

## Sample Output

**show interfaces lsi extensive**

user@host> **show interfaces lsi extensive**

Physical interface: lsi

Logical interface lsi.84934656 (Index 363) (SNMP ifIndex 586) (Generation 194)

Flags: Up Point-To-Point SNMP-Traps 0x4000000 Encapsulation: LSI-NULL

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

Protocol vpls, MTU: Unlimited, Generation: 279, Route table: 10

Logical interface lsi.84934657 (Index 366) (SNMP ifIndex 589) (Generation 197)

Flags: Up Point-To-Point SNMP-Traps 0x4000000 Encapsulation: LSI-NULL

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
Protocol vpls, MTU: Unlimited, Generation: 282, Route table: 10
```

## show l2circuit connections

### Syntax

```
show l2circuit connections
<brief | extensive | summary>
<down | up | up-down>
<history>
<interface interface-name>
<logical-system (all | logical-system-name)>
<neighbor neighbor>
<status>
```

### Release Information

Command introduced before Junos OS Release 7.4.

Display enhancements in Junos OS Release 9.6.

Display enhancements in Junos OS Release 10.2.

Display enhancements in Junos OS Release 12.1.

Display enhancements in Junos OS Release 13.2.

Statement introduced in Junos OS Release 14.1X53-D10 for the QFX Series and for EX4600 switches.

### Description

Display status information about Layer 2 virtual circuits from the local provider edge (PE) router to its neighbors.

### Options

**none**—Display standard information about Layer 2 virtual circuits on all interfaces for all neighbors.

**brief | extensive | summary**—(Optional) Display the specified level of output. Use history to display information about connection history. Use status to display information about the connection and interface status.

**down | up | up-down**—(Optional) Display nonoperational, operational, or both kinds of connections.

**history**—(Optional) Display information about connection history.

**interface *interface-name***—(Optional) Show all Layer 2 virtual circuits on an interface.

**logical-system (all | *logical-system-name*)**—(Optional) Perform this operation on all logical systems or on a particular logical system.

**neighbor *neighbor***—(Optional) IP address of a specific neighbor.

**status**—(Optional) Display information about the connection and interface status.

### Required Privilege Level

view

List of Sample Output

- [show l2circuit connections on page 305](#)
- [show l2circuit connections interface on page 306](#)
- [show l2circuit connections extensive on page 307](#)
- [show l2circuit connections extensive \(Pseudowire Redundancy with Hot Standby\) on page 308](#)

Output Fields

[Table 8 on page 300](#) lists the output fields for the **show l2circuit connections** command. Output fields are listed in the approximate order in which they appear.

Table 8: show l2circuit connections Output Fields

Field Name	Field Description
Layer-2 Circuit Connections	Displays the legends for connection and interface status.
Neighbor	Remote PE neighbor.
Interface	Logical PE-to-CE interface on which the virtual circuit is configured.
Type	VC type: <b>rmt</b> (remote) or <b>loc</b> (local).

Table 8: show l2circuit connections Output Fields *(continued)*

Field Name	Field Description
Legend for connection status (St)	

Table 8: show l2circuit connections Output Fields (continued)

Field Name	Field Description
	<p>Status of the virtual circuit connection:</p> <ul style="list-style-type: none"> <li>• <b>EI</b>—The local virtual circuit interface is configured with an encapsulation that is not supported.</li> <li>• <b>MM</b>—The two routers do not agree on an MTU value, which causes an MTU mismatch.</li> <li>• <b>EM</b>—The encapsulation type received on this virtual circuit from the neighbor does not match the local virtual circuit interface encapsulation type.</li> <li>• <b>CM</b>—The two routers do not agree on a control word, which causes a control word mismatch.</li> <li>• <b>VM</b>—The remote and local VLAN IDs do not match across the Layer 2 circuit.</li> <li>• <b>OL</b>—No advertisement has been received for this virtual circuit from the neighbor. There is no outgoing label available for use by this virtual circuit.</li> <li>• <b>NC</b>—The interface is not configured as a CCC or TCC interface.</li> <li>• <b>BK</b>—The virtual circuit has switched to a backup connection.</li> <li>• <b>CB</b>—The remote PE router is advertising a different cell bundle from that configured on the local PE router.</li> <li>• <b>LD</b>—The connection to the local site is signaled down, because the CE-facing interface to the local site is down.</li> <li>• <b>RD</b>—The remote neighbor is down. It has signaled a problem using the pseudowire status code.</li> <li>• <b>NP</b>—The router detects that interface hardware is not present. The hardware may be offline, a PIC may not be of the desired type, or the interface may be configured in a different routing instance.</li> <li>• <b>Dn</b>—The virtual circuit is down.</li> <li>• <b>VC-Dn</b>—The virtual circuit is down because there is no tunnel LSP from the local PE router to the neighbor.</li> <li>• <b>UP</b>—The virtual circuit is operational.</li> <li>• <b>CF</b>—The router cannot find enough bandwidth to the remote router to satisfy the Layer 2 circuit bandwidth requirement.</li> <li>• <b>IB</b>—The bit rate is incompatible for Time Division Multiplexing (TDM).</li> <li>• <b>TDM</b>—TDM is not configured correctly.</li> <li>• <b>ST</b>—The virtual circuit has been switched to a standby connection.</li> <li>• <b>SP</b>—The virtual circuit connection is using a static pseudowire.</li> <li>• <b>RS</b>—The remote site is in a standby state.</li> </ul>

Table 8: show l2circuit connections Output Fields (continued)

Field Name	Field Description
	<ul style="list-style-type: none"> <li>• <b>XX</b>—The virtual circuit is down for an unknown reason. This is a programming error.</li> </ul>
<b>Time last up</b>	Date and time the virtual circuit was last operational.
<b># Up trans</b>	Number of times the virtual circuit came up.
<i>local-interface-name</i>	Name of the local PE-to-CE interface.
<b>Status</b>	Status of the local interface.
<b>Up</b>	Interface is operational.
<b>Dn</b>	Interface is not operational.
<b>NP</b>	Not present. Interface does not exist.
<b>DS</b>	Disabled. Interface has been administratively disabled.
<b>WE</b>	Wrong encapsulation. The interface is not configured as CCC.
<b>UN</b>	Interface status is initialized.
<b>Encapsulation</b>	Encapsulation of the local interface.
<b>Flow Label Transmit</b>	Flow label transmit status.
<b>Flow Label Receive</b>	Flow label receive status.
<b>Remote PE</b>	Prefix of the remote PE router.
<b>Negotiated control-word</b>	Whether the use of the control word has been negotiated for this virtual circuit: <b>Yes (Null)</b> or <b>No</b> .
<b>Incoming label</b>	Label used by the remote side of the virtual circuit to send packets destined to the local side. This label is routed to the local virtual circuit interface.

Table 8: show l2circuit connections Output Fields (continued)

Field Name	Field Description
<b>Outgoing label</b>	Label used by the local side of the virtual circuit to send packets to the remote side of the virtual circuit. Packets originated on the local virtual circuit interface are encapsulated with this label before being placed on the tunnel LSP to the neighbor for this virtual circuit. This label is allocated by the neighbor and is used in demultiplexing incoming packets destined for this virtual circuit.
<b>Negotiated PW status TLV</b>	Displays the pseudowire status type, length, and value (TLV). TLVs are a method of encoding variable-length or optional information. If the pseudowire status TLV is used, the corresponding local or neighbor PE router status code is also displayed.
<b>local PW status code</b>	If the pseudowire status TLV is used, displays the local PE router status code.
<b>Neighbor PW status code</b>	If the pseudowire status TLV is used, displays the neighbor PE router status code.
<b>Local interface</b>	Name of the local interface used for the Layer 2 circuit connection.
<b>Status</b>	Status of the local interface ( <b>Up</b> or <b>Down</b> ).
<b>Encapsulation</b>	Encapsulation configured for the local interface.
<b>APS-active</b>	Indicates that the interface belongs to the working circuit.
<b>APS-inactive</b>	Indicates that the interface belongs to the protect circuit.
<b>Connection protection</b>	Whether or not connection protection is configured for the Layer 2 circuit to the neighbor: <b>Yes</b> or <b>No</b> .
<b>VC bandwidth</b>	Bandwidth requirement of the Layer 2 circuit.
<b>Time</b>	Time at which the event occurred.



Table 8: show l2circuit connections Output Fields (continued)

Field Name	Field Description
<b>Connection History</b>	<p>Event types logged in history.</p> <ul style="list-style-type: none"> <li>• <b>loc intf up</b>—Local virtual circuit interface went up.</li> <li>• <b>loc intf down</b>—Local virtual circuit interface went down.</li> <li>• <b>In lbl Update</b>—Incoming label has been updated.</li> <li>• <b>Out lbl Update</b>—Outgoing label has been updated.</li> <li>• <b>PE route changed</b>—Route to PE router has been updated.</li> <li>• <b>PE route down</b>—Route to PE router is down.</li> <li>• <b>rmt side marked</b>—Remote side is marked.</li> <li>• <b>VC Dn</b>—Remote side indicated that its end of the virtual circuit is down (if the tunnel LSP from the remote side to the local side is down).</li> <li>• <b>status update timer</b>—Status update timer processing. It computes the state of the virtual circuit, and determines whether it should be advertised to or withdrawn from the remote side.</li> </ul>

## Sample Output

### show l2circuit connections

user@host> show l2circuit connections

```

Layer-2 Circuit Connections:

Legend for connection status (St)
EI -- encapsulation invalid      NP -- interface h/w not present
MM -- mtu mismatch              Dn -- down
EM -- encapsulation mismatch    VC-Dn -- Virtual circuit Down
CM -- control-word mismatch     Up -- operational
VM -- vlan id mismatch         CF -- Call admission control failure
OL -- no outgoing label        IB -- TDM incompatible bitrate
NC -- intf encaps not CCC/TCC  TM -- TDM misconfiguration
BK -- Backup Connection        ST -- Standby Connection
CB -- rcvd cell-bundle size bad SP -- Static Pseudowire
LD -- local site signaled down RS -- remote site standby
RD -- remote site signaled down HS -- hot standby
XX -- unknown

Legend for interface status

```

```

Up -- operational
Dn -- down
Neighbor: 10.255.245.51
  Interface                Type  St      Time last up          # Up trans
  ge-2/0/2.600(vc 5)      rmt   Up      Dec  7 18:11:18 2009      1
  Remote PE: 10.255.245.51, Negotiated control-word: No
  Incoming label: 299856, Outgoing label: 299808
  Negotiated PW status TLV: No
  Local interface: ge-2/0/2.600, Status: Up, Encapsulation: VLAN
  Flow Label Transmit: No, Flow Label Receive: No
  Auto-sensed or Programmed by XYZ

```

## Sample Output

### show l2circuit connections interface

```
user@host> show l2circuit connections interface t1-2/0/0:1:1.0
```

```

Layer-2 Circuit Connections:

Legend for connection status (St)
EI -- encapsulation invalid      NP -- interface h/w not present
MM -- mtu mismatch              Dn -- down
EM -- encapsulation mismatch     VC-Dn -- Virtual circuit Down
CM -- control-word mismatch     Up -- operational
VM -- vlan id mismatch          CF -- Call admission control failure
OL -- no outgoing label         IB -- TDM incompatible bitrate
NC -- intf encaps not CCC/TCC   TM -- TDM misconfiguration
BK -- Backup Connection         ST -- Standby Connection
CB -- rcvd cell-bundle size bad SP -- Static Pseudowire
LD -- local site signaled down  RS -- remote site standby
RD -- remote site signaled down HS -- hot standby
XX -- unknown

Legend for interface status
Up -- operational
Dn -- down
Neighbor: 10.1.1.1
  Interface                Type  St      Time last up          # Up trans
  t1-2/0/0:1:1.0(vc 1)(SP) rmt   Up      Apr 27 04:21:02 2011      1
  Remote PE: 10.1.1.1, Negotiated control-word: Yes (Non-null)
  Incoming label: 1010001, Outgoing label: 1000001

```

```

Negotiated PW status TLV: No
Local interface: t1-1/0/0:1:1.0, Status: Up, Encapsulation: SATOP-T1,
APS-active
Local interface: t1-2/0/0:1:1.0, Status: Up, Encapsulation: SATOP-T1,
APS-inactive

```

## Sample Output

**show l2circuit connections extensive**

user@host>**show l2circuit connections extensive**

```

Layer-2 Circuit Connections:

Legend for connection status (St)
EI -- encapsulation invalid      NP -- interface h/w not present
MM -- mtu mismatch              Dn -- down
EM -- encapsulation mismatch     VC-Dn -- Virtual circuit Down
CM -- control-word mismatch     Up -- operational
VM -- vlan id mismatch          CF -- Call admission control failure
OL -- no outgoing label         IB -- TDM incompatible bitrate
NC -- intf encaps not CCC/TCC   TM -- TDM misconfiguration
BK -- Backup Connection         ST -- Standby Connection
CB -- rcvd cell-bundle size bad SP -- Static Pseudowire
LD -- local site signaled down  RS -- remote site standby
RD -- remote site signaled down HS -- hot standby
XX -- unknown

Legend for interface status
Up -- operational
Dn -- down
Neighbor: 10.255.49.149

```

Interface	Type	St	Time last up	# Up trans
ae0.0(vc 100)	rmt	Up	Aug 31 09:36:12 2009	1

```

Remote PE: 10.255.49.149, Negotiated control-word: Yes (Null)
Incoming label: 299824, Outgoing label: 299776
Negotiated PW status TLV: Yes
local PW status code: 0x00000000, Neighbor PW status code: 0x00000000
Local interface: ae0.0, Status: Up, Encapsulation: ETHERNET
Connection protection: Yes
Connection History:
Aug 31 09:36:12 2009 status update timer

```

```

Aug 31 09:36:12 2009 PE route changed
Aug 31 09:36:12 2009 Out lbl Update 299776
Aug 31 09:36:12 2009 In lbl Update 299824
Aug 31 09:36:12 2009 loc intf up ae0.0

```

## Sample Output

### show l2circuit connections extensive (Pseudowire Redundancy with Hot Standby)

user@host>show l2circuit connections extensive

```

Layer-2 Circuit Connections:

Legend for connection status (St)
EI -- encapsulation invalid      NP -- interface h/w not present
MM -- mtu mismatch              Dn -- down
EM -- encapsulation mismatch     VC-Dn -- Virtual circuit Down
CM -- control-word mismatch     Up -- operational
VM -- vlan id mismatch          CF -- Call admission control failure
OL -- no outgoing label         IB -- TDM incompatible bitrate
NC -- intf encaps not CCC/TCC   TM -- TDM misconfiguration
BK -- Backup Connection         ST -- Standby Connection
CB -- rcvd cell-bundle size bad SP -- Static Pseudowire
LD -- local site signaled down  RS -- remote site standby
RD -- remote site signaled down HS -- Hot-standby Connection
XX -- unknown

Legend for interface status
Up -- operational
Dn -- down
Neighbor: 192.0.2.101
Interface                Type  St    Time last up          # Up trans
ge-1/3/2.600(vc 1)      rmt   Up    Jan 24 11:00:26 2013      1
Remote PE: 192.0.2.101, Negotiated control-word: Yes (Null)
Incoming label: 299776, Outgoing label: 299776
Negotiated PW status TLV: Yes
local PW status code: 0x00000000, Neighbor PW status code: 0x00000000
Local interface: ge-1/3/2.600, Status: Up, Encapsulation: VLAN
Connection History:
Jan 24 11:00:26 2013 status update timer
Jan 24 11:00:26 2013 PE route changed
Jan 24 11:00:26 2013 Out lbl Update 299776

```

```

Jan 24 11:00:26 2013 In lbl Update                299776
Jan 24 11:00:26 2013 loc intf up                  ge-1/3/2.600
Neighbor: 192.0.2.102
Interface                Type  St      Time last up      # Up trans
ge-1/3/2.600(vc 2)      rmt   HS      -----          ----
Remote PE: 192.0.2.102, Negotiated control-word: Yes (Null)
Incoming label: 299792, Outgoing label: 299776
Negotiated PW status TLV: Yes
local PW status code: 0x00000020, Neighbor PW status code: 0x00000000
Local interface: ge-1/3/2.600, Status: Up, Encapsulation: VLAN

```

## show l2vpn connections

### Syntax

```
show l2vpn connections
<brief | extensive>
<down | up | up-down>
<history>
<instance instance>
<instance-history>
<local-site local-site>
<logical-system (all | logical-system-name)>
<remote-site remote-site>
<status>
<summary>
```

### Release Information

Command introduced before Junos OS Release 7.4.

**instance-history** option introduced in Junos OS Release 12.3R2.

### Description

Display Layer 2 virtual private network (VPN) connections.

### Options

**none**—Display all Layer 2 VPN connections for all routing instances.

**brief | extensive**—(Optional) Display the specified level of output.

**down | up | up-down**—(Optional) Display nonoperational, operational, or both kinds of connections.

**history**—(Optional) Display information about connection history.

**instance *instance***—(Optional) Display connections for the specified routing instance only.

**instance-history**—(Optional) Display information about connection history for a particular instance.

**local-site *local-site***—(Optional) Display connections for the specified Layer 2 VPN local site name or ID only.

**logical-system (all | *logical-system-name*)**—(Optional) Perform this operation on all logical systems or on a particular logical system.

**remote-site *remote-site***—(Optional) Display connection for the specified Layer 2 VPN remote site ID only.

**status**—(Optional) Display information about the connection and interface status.

**summary**—(Optional) Display summary of all Layer 2 VPN connections information.

## Required Privilege Level

view

## List of Sample Output

[show l2vpn connections on page 314](#)

[show l2vpn connections on page 315](#)

[show l2vpn connections extensive on page 316](#)

[show l2vpn connections extensive \(VPWS\) on page 317](#)

## Output Fields

[Table 9 on page 311](#) lists the output fields for the **show l2vpn connections** command. Output fields are listed in the approximate order in which they appear.

Table 9: show l2vpn connections Output Fields

Field Name	Field Description
Instance	Name of Layer 2 VPN instance.
L2vpn-id	For BGP autodiscovery, a globally unique Layer 2 VPN community identifier for the instance.
Local-ID	BGP <b>local-address</b> assigned to the local routing device.
Local site	Name of local site.
Local source-attachment-id	For FEC 129, the VPWS source attachment identifier. The point-to-point nature of VPWS requires that you specify the source access individual identifier (SAII) and the target access individual identifier (TAII). This SAII-TAII pair defines a unique pseudowire between two PE devices.
Target-attachment-id	For FEC 129, the VPWS target attachment identifier. If the configured target identifier matches a source identifier advertised by a remote PE device by way of a BGP auto-discovery message, the pseudowire between that source-target pair is signaled. If there is no match between an advertised source identifier and the configured target identifier, the pseudowire is not established.
Interface name	Name of interface.
Remote Site ID	Remote site ID.
Label Offset	Numbers within the label block that are skipped to find the next label base.

Table 9: show l2vpn connections Output Fields (continued)

Field Name	Field Description
<b>Label-base</b>	Advertises the first label in a block of labels. A remote PE router uses this first label when sending traffic toward the advertising PE router.
<b>Range</b>	Advertises the label block size.
<b>status-vector</b>	Bit vector advertising the state of local PE-CE circuits to remote PE routers. A bit value of <b>0</b> indicates that the local circuit and LSP tunnel to the remote PE router are up, whereas a value of <b>1</b> indicates either one or both are down.
<b>connection-site</b>	Name of the connection site.
<b>Type</b>	Type of connection: <b>loc</b> (local) or <b>rmt</b> (remote).
<b>St</b>	Status of the connection. (For a list of possible values, see the <b>Legend for connection status (St)</b> field.)
<b>Time last up</b>	Time that the connection was last in the <b>Up</b> condition.
<b># Up trans</b>	Number of transitions from <b>Down</b> to <b>Up</b> condition.
<b>Local circuit</b>	Address and status of local circuit.
<b>Remote circuit</b>	Address and status of remote circuit.
<b>St</b>	<p>Status of the Layer 2 VPN connection (corresponds with Legend for Connection Status):</p> <ul style="list-style-type: none"> <li>• <b>EI</b>—The local Layer 2 VPN interface is configured with an encapsulation that is not supported.</li> <li>• <b>EM</b>—The encapsulation type received on this Layer 2 VPN connection from the neighbor does not match the local Layer 2 VPN connection interface encapsulation type.</li> <li>• <b>VC-Dn</b>—The virtual circuit is currently down.</li> <li>• <b>CM</b>—The two routers do not agree on a control word, which causes a control word mismatch.</li> <li>• <b>CN</b>—The virtual circuit is not provisioned properly.</li> <li>• <b>OR</b>—The label associated with the virtual circuit is out of range.</li> <li>• <b>OL</b>—No advertisement has been received for this virtual circuit from the neighbor. There is no outgoing label available for use by this virtual circuit.</li> </ul>



Table 9: show l2vpn connections Output Fields (continued)

Field Name	Field Description
	<ul style="list-style-type: none"> <li>• <b>LD</b>—All of the CE-facing interfaces to the local site are down. Therefore, the connection to the local site is signaled as down to the other PE routers. No pseudowires can be established.</li> <li>• <b>RD</b>—All the interfaces to the remote neighbor are down. Therefore, the remote site has been signaled as down to the other PE routers. No pseudowires can be established.</li> <li>• <b>LN</b>—The local site has lost path selection to the remote site and therefore no pseudowires can be established from this local site.</li> <li>• <b>RN</b>—The remote site has lost path selection to a local site or other remote site and therefore no pseudowires are established to this remote site.</li> <li>• <b>XX</b>—The Layer 2 VPN connection is down for an unknown reason. This is a programming error.</li> <li>• <b>NC</b>—The interface encapsulation is not configured as an appropriate CCC, TCC, or Layer 2 VPN encapsulation.</li> <li>• <b>WE</b>—The encapsulation configured for the interface does not match the encapsulation configured for the associated connection within the Layer 2 VPN routing instance.</li> <li>• <b>NP</b>—The router detects that interface hardware is not present. The hardware might be offline, a PIC might not be of the desired type, or the interface might be configured in a different routing instance.</li> <li>• <b>-&gt;</b>—Only the outbound connection is up.</li> <li>• <b>&lt;-</b>—Only the inbound connection is up.</li> <li>• <b>Up</b>—The Layer 2 VPN connection is operational.</li> <li>• <b>Dn</b>—The Layer 2 VPN connection is down.</li> <li>• <b>CF</b>—The router cannot find enough bandwidth to the remote router to satisfy the Layer 2 VPN connection bandwidth requirement.</li> <li>• <b>SC</b>—The local site identifier matches the remote site identifier. No pseudowire can be established between these two sites. You should configure different values for the local and remote site identifiers.</li> <li>• <b>LM</b>—The local site identifier is not the minimum designated, meaning it is not the lowest. There is another local site with a lower site identifier. Pseudowires are not being established to this local site, and the associated local site identifier is not being used to distribute Layer 2 VPN label blocks. However, this is not an error state. Traffic continues to be forwarded to the PE router interfaces connected to the local sites when the local sites are in this state.</li> <li>• <b>RM</b>—The remote site identifier is not the minimum designated, meaning it is not the lowest. There is another remote site connected to the same</li> </ul>

Table 9: show l2vpn connections Output Fields (continued)

Field Name	Field Description
	<p>PE router which has lower site identifier. The PE router cannot established a pseduowire to this remote site and the associated remote site identifier cannot be used to distribute VPLS label blocks. However, this is not an error state. Traffic can continue to be forwarded to the PE router interface connected to this remote site when the remote site is in this state.</p> <ul style="list-style-type: none"> <li>• <b>IL</b>—The incoming packets for the Layer 2 VPN connection have no MPLS label.</li> </ul>
<b>Remote PE</b>	Address of the remote provider edge router.
<b>Incoming label</b>	Name of the incoming label.
<b>Outgoing label</b>	Name of the outgoing label.
<b>Egress Protection</b>	Whether the given PVC is protected by connection protection logic using egress protection for BGP signaled layer 2 services.
<b>Flow Label Receive</b>	Capability to pop the flow label in the receive direction to the remote provider edge (PE) router
<b>Flow Label Transmit</b>	Capability to push the flow label in the transmit direction to the provider edge (PE) router
<b>Time</b>	Date and time of Layer 2 VPN connection event.
<b>Event</b>	Type of event.
<b>Interface/Lbl/PE</b>	Interface, label, or PE router.

## Sample Output

show l2vpn connections

```
user@host> show l2vpn connections
```

```
L2VPN Connections :
Instance : vpna
Edge protection: Not-Primary
```

```

Local site: 2 (ce-2)
offset: 1, range: 3, label-base: 32768
      connection-site      Type  St   Time last up      # Up trans
3 (3)                      loc   Up   Jul 18 20:45:46 2001      1
    Local circuit: fe-0/0/0.1, Status: Up
    Remote circuit: fe-0/0/3.0, Status: Up
1                          rmt   Up   Jul 18 21:47:25 2001      1
    Local circuit: fe-0/0/0.0, Status: Up
    Remote PE: 192.0.2.1
    Incoming label: 32768, Outgoing label: 32769
Local site: 3 (ce-3)
offset: 1, range: 2, label-base: 33792
      connection-site      Type  St   Time last up      # Up trans
2 (ce-b)                   loc   Up   Jul 18 20:45:46 2001      1
    Local circuit: fe-0/0/0.1, Status: Up
    Remote circuit: fe-0/0/3.0, Status: Up
1                          rmt   Up   Jul 18 21:47:25 2001      1
    Local circuit: fe-0/0/3.1, Status: Up
    Remote PE: 192.0.2.1
    Incoming label: 33792, Outgoing label: 32770

```

## show l2vpn connections

user@host> show l2vpn connections

Layer-2 VPN connections:

Legend for connection status (St)

EI -- encapsulation invalid	NC -- interface encapsulation not CCC/TCC/VPLS
EM -- encapsulation mismatch	WE -- interface and instance encaps not same
VC-Dn -- Virtual circuit down	NP -- interface hardware not present
CM -- control-word mismatch	-> -- only outbound connection is up
CN -- circuit not provisioned	<- -- only inbound connection is up
OR -- out of range	Up -- operational
OL -- no outgoing label	Dn -- down
LD -- local site signaled down	CF -- call admission control failure
RD -- remote site signaled down	SC -- local and remote site ID collision
LN -- local site not designated	LM -- local site ID not minimum designated
RN -- remote site not designated	RM -- remote site ID not minimum designated
XX -- unknown connection status	IL -- no incoming label
MM -- MTU mismatch	MI -- Mesh-Group ID not available
BK -- Backup connection	ST -- Standby connection
PF -- Profile parse failure	PB -- Profile busy

```

RS -- remote site standby      SN -- Static Neighbor
LB -- Local site not best-site  RB -- Remote site not best-site
VM -- VLAN ID mismatch

Legend for interface status
Up -- operational
Dn -- down

Instance: l2vpn-inst
Edge protection: Not-Primary
  Local site: pe2 (2)
    connection-site      Type  St      Time last up      # Up trans
    1                    rmt   Up      Jun 22 14:46:50 2015      1
    Remote PE: 10.255.255.1, Negotiated control-word: Yes (Null)
    Incoming label: 800002, Outgoing label: 800003
    Local interface: ge-0/0/1.300, Status: Up, Encapsulation: VLAN
    Flow Label Transmit: Yes, Flow Label Receive: Yes

```

### show l2vpn connections extensive

```
user@host> show l2vpn connections extensive
```

```

L2VPN Connections:
Instance: vpn-a
Edge protection: Not-Primary
Local site: ce-a (1)
  Interface name      Remote Site ID
  fe-0/0/0.0          2
  Label Offset      Offset      Range
    32768            1          2
  connection-site      Type  St      Time last up      # Up trans
  2                    rmt   Up      Aug  3 00:08:14 2001      1
  Local circuit: fe-0/0/0.0, Status: Up
  Remote PE: 192.168.24.1
  Incoming label: 32769, Outgoing label: 32768
  Egress Protection: Yes
    Time              Event              Interface/Lbl/PE
  Aug  3 00:08:14 2001 PE route up
  Aug  3 00:08:14 2001 Out lbl Update      32768
  Aug  3 00:08:14 2001 In lbl Update      32769
  Aug  3 00:08:14 2001 ckt0 up            fe-0/0/0.0

```

**show l2vpn connections extensive (VPWS)**

```
user@host> show l2vpn connections
```

```
Layer-2 VPN connections:
```

```
Legend for connection status (St)
```

```

EI -- encapsulation invalid      NC -- interface encapsulation not CCC/TCC/VPLS
EM -- encapsulation mismatch     WE -- interface and instance encaps not same
VC-Dn -- Virtual circuit down   NP -- interface hardware not present
CM -- control-word mismatch     -> -- only outbound connection is up
CN -- circuit not provisioned    <- -- only inbound connection is up
OR -- out of range              Up -- operational
OL -- no outgoing label         Dn -- down
LD -- local site signaled down   CF -- call admission control failure
RD -- remote site signaled down  SC -- local and remote site ID collision
LN -- local site not designated  LM -- local site ID not minimum designated
RN -- remote site not designated RM -- remote site ID not minimum designated
XX -- unknown connection status IL -- no incoming label
MM -- MTU mismatch              MI -- Mesh-Group ID not available
BK -- Backup connection         ST -- Standby connection
PF -- Profile parse failure     PB -- Profile busy
RS -- remote site standby       SN -- Static Neighbor
LB -- Local site not best-site  RB -- Remote site not best-site
VM -- VLAN ID mismatch

```

```
Legend for interface status
```

```

Up -- operational
Dn -- down

```

```
Instance: FEC129-VPWS
```

```
L2vpn-id: 100:100
```

```
Number of local interfaces: 1
```

```
Number of local interfaces up: 1
```

```
ge-2/0/5.0
```

```
Local source-attachment-id: 1 (ONE)
```

Target-attachment-id	Type	St	Time last up	# Up trans
2	rmt	Up	Nov 28 16:16:14 2012	1

```
Remote PE: 198.51.100.2, Negotiated control-word: No
```

```
Incoming label: 299792, Outgoing label: 299792
```

```
Local interface: ge-2/0/5.0, Status: Up, Encapsulation: ETHERNET
```

```
Connection History:
```

```
Nov 28 16:16:14 2012 status update timer
```

```
Nov 28 16:16:14 2012 PE route changed
```

```
Nov 28 16:16:14 2012 Out lbl Update 299792
```

Nov 28 16:16:14 2012	In lbl Update	299792
Nov 28 16:16:14 2012	loc intf up	ge-2/0/5.0

## show pim snooping interfaces

### Syntax

```
show pim snooping interfaces
<brief | detail>
<instance instance-name>
<interface interface-name>
<logical-system logical-system-name>
<vlan-id vlan-identifier>
```

### Release Information

Command introduced in Junos OS Release 12.3 for MX Series 5G Universal Routing Platforms.

Command introduced in Junos OS Release 13.2 for M Series Multiservice Edge devices.

### Description

Display information about PIM snooping interfaces.

### Options

**none**—Display detailed information.

**brief | detail**—(Optional) Display the specified level of output.

**instance** *<instance-name>*—(Optional) Display PIM snooping interface information for the specified routing instance.

**interface** *<interface-name>*—(Optional) Display PIM snooping information for the specified interface only.

**logical-system** *logical-system-name*—(Optional) Display information about a particular logical system, or type 'all'.

**vlan-id** *<vlan-identifier>*—(Optional) Display PIM snooping interface information for the specified VLAN.

### Required Privilege Level

view

### RELATED DOCUMENTATION

| *PIM Snooping for VPLS*

### List of Sample Output

[show pim snooping interfaces on page 320](#)

[show pim snooping interfaces instance vpls1 on page 321](#)

[show pim snooping interfaces interface <interface-name> on page 321](#)

[show pim snooping interfaces vlan-id <vlan-id> on page 322](#)

### Output Fields

[Table 10 on page 320](#) lists the output fields for the **show pim snooping interface** command. Output fields are listed in the approximate order in which they appear.

**Table 10: show pim snooping interface Output Fields**

Field Name	Field Description	Level of Output
<b>Instance</b>	Routing instance for PIM snooping.	All levels
<b>Learning-Domain</b>	Learning domain for snooping.	All levels
<b>Name</b>	Router interfaces that are part of this learning domain.	All levels
<b>State</b>	State of the interface: <b>Up</b> , or <b>Down</b> .	All levels
<b>IP-Version</b>	Version of IP used: <b>4</b> for IPv4, or <b>6</b> for IPv6.	All levels
<b>NbrCnt</b>	Number of neighboring routers connected through the specified interface.	All levels
<b>DR address</b>	IP address of the designated router.	All levels

## Sample Output

**show pim snooping interfaces**

user@host> **show pim snooping interfaces**

```

Instance: vpls1
Learning-Domain: vlan-id 10
Name State IP-Version NbrCnt
ge-1/3/1.10 Up 4 1
ge-1/3/3.10 Up 4 1
ge-1/3/5.10 Up 4 1
ge-1/3/7.10 Up 4 1
DR address: 192.0.2.5
DR flooding is ON

Learning-Domain: vlan-id 20
Name State IP-Version NbrCnt
ge-1/3/1.20 Up 4 1

```



```

ge-1/3/3.20 Up 4 1
ge-1/3/5.20 Up 4 1
ge-1/3/7.20 Up 4 1
DR address: 192.0.2.6
DR flooding is ON

```

### show pim snooping interfaces instance vpls1

user@host> show pim snooping interfaces instance vpls1

```

Instance: vpls1

Learning-Domain: vlan-id 10
Name State IP-Version NbrCnt
ge-1/3/1.10 Up 4 1
ge-1/3/3.10 Up 4 1
ge-1/3/5.10 Up 4 1
ge-1/3/7.10 Up 4 1
DR address: 192.0.2.5
DR flooding is ON

Learning-Domain: vlan-id 20
Name State IP-Version NbrCnt
ge-1/3/1.20 Up 4 1
ge-1/3/3.20 Up 4 1
ge-1/3/5.20 Up 4 1
ge-1/3/7.20 Up 4 1
DR address: 192.0.2.6
DR flooding is ON

```

### show pim snooping interfaces interface <interface-name>

user@host> show pim snooping interfaces interface ge-1/3/1.10

```

Instance: vpls1
Learning-Domain: vlan-id 10

Name State IP-Version NbrCnt
ge-1/3/1.10 Up 4 1
DR address: 192.0.2.5
DR flooding is ON

Learning-Domain: vlan-id 20

```

```
DR address: 192.0.2.6  
DR flooding is ON
```

**show pim snooping interfaces vlan-id <vlan-id>**

user@host> **show pim snooping interfaces vlan-id 10**

```
Instance: vpls1  
Learning-Domain: vlan-id 10  
  
Name State IP-Version NbrCnt  
ge-1/3/1.10 Up 4 1  
ge-1/3/3.10 Up 4 1  
ge-1/3/5.10 Up 4 1  
ge-1/3/7.10 Up 4 1  
DR address: 192.0.2.5  
DR flooding is ON
```

## show pim snooping join

### Syntax

```
show pim snooping join
<brief | detail | extensive>
<instance instance-name>
<logical-system logical-system-name>
<vlan-id vlan-id>
```

### Release Information

Command introduced in Junos OS Release 12.3 for MX Series 5G Universal Routing Platforms.

Command introduced in Junos OS Release 13.2 for M Series Multiservice Edge devices.

### Description

Display information about Protocol Independent Multicast (PIM) snooping joins.

### Options

**none**—Display detailed information.

**brief | detail | extensive**—(Optional) Display the specified level of output.

**instance *instance-name***—(Optional) Display PIM snooping join information for the specified routing instance.

**logical-system *logical-system-name***—(Optional) Display information about a particular logical system, or type 'all'.

**vlan-id *vlan-identifier***—(Optional) Display PIM snooping join information for the specified VLAN.

### Required Privilege Level

view

### RELATED DOCUMENTATION

| [PIM Snooping for VPLS](#)

### List of Sample Output

[show pim snooping join on page 325](#)

[show pim snooping join extensive on page 326](#)

[show pim snooping join instance on page 326](#)

[show pim snooping join vlan-id on page 327](#)

### Output Fields

Table 11 on page 324 lists the output fields for the **show pim snooping join** command. Output fields are listed in the approximate order in which they appear.

**Table 11: show pim snooping join Output Fields**

Field Name	Field Description	Level of Output
<b>Instance</b>	Routing instance for PIM snooping.	All levels
<b>Learning-Domain</b>	Learning domain for PIM snooping.	All levels
<b>Group</b>	Multicast group address.	All levels
<b>Source</b>	Multicast source address: <ul style="list-style-type: none"> <li>• * (wildcard value)</li> <li>• &lt;ipv4-address&gt;</li> <li>• &lt;ipv6-address&gt;</li> </ul>	All levels
<b>Flags</b>	PIM flags: <ul style="list-style-type: none"> <li>• <b>bidirectional</b>—Bidirectional mode entry.</li> <li>• <b>dense</b>—Dense mode entry.</li> <li>• <b>rptree</b>—Entry is on the rendezvous point tree.</li> <li>• <b>sparse</b>—Sparse mode entry.</li> <li>• <b>spt</b>—Entry is on the shortest-path tree for the source.</li> <li>• <b>wildcard</b>—Entry is on the shared tree.</li> </ul>	All levels
<b>Upstream state</b>	Information about the upstream interface: <ul style="list-style-type: none"> <li>• <b>Join to RP</b>—Sending a join to the rendezvous point.</li> <li>• <b>Join to Source</b>—Sending a join to the source.</li> <li>• <b>Local RP</b>—Sending neither join messages nor prune messages toward the RP, because this router is the rendezvous point.</li> <li>• <b>Local Source</b>—Sending neither join messages nor prune messages toward the source, because the source is locally attached to this routing device.</li> <li>• <b>Prune to RP</b>—Sending a prune to the rendezvous point.</li> <li>• <b>Prune to Source</b>—Sending a prune to the source.</li> </ul> <p>NOTE: RP group range entries have <b>None</b> in the <b>Upstream state</b> field because RP group ranges do not trigger actual PIM join messages between routers.</p>	All levels

Table 11: show pim snooping join Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Upstream neighbor</b>	Information about the upstream neighbor: <b>Direct</b> , <b>Local</b> , <b>Unknown</b> , or a specific IP address.  For bidirectional PIM, <b>Direct</b> means that the interface is directly connected to a subnet that contains a phantom RP address.	All levels
<b>Upstream port</b>	RPF interface toward the source address for the source-specific state (S,G) or toward the rendezvous point (RP) address for the non-source-specific state (*,G).  For bidirectional PIM, <b>RP Link</b> means that the interface is directly connected to a subnet that contains a phantom RP address.	All levels
<b>Downstream port</b>	Information about downstream interfaces.	<b>extensive</b>
<b>Downstream neighbors</b>	Address of the downstream neighbor.	<b>extensive</b>
<b>Timeout</b>	Time remaining until the downstream join state is updated (in seconds).	<b>extensive</b>

## Sample Output

show pim snooping join

user@host> show pim snooping join

```
Instance: vpls1

Learning-Domain: vlan-id 10
Group: 198.51.100.2
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 192.0.2.4, port: ge-1/3/5.10
```

```
Learning-Domain: vlan-id 20
Group: 198.51.100.3
Source: *
Flags: sparse,rptree,wildcard
```

```
Upstream state: None
Upstream neighbor: 203.0.113.4, port: ge-1/3/5.20
```

### show pim snooping join extensive

```
user@host> show pim snooping join extensive
```

```
Instance: vpls1
Learning-Domain: vlan-id 10

Group: 198.51.100.2
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 192.0.2.4, port: ge-1/3/5.10
Downstream port: ge-1/3/1.10
Downstream neighbors:
192.0.2.2 State: Join Flags: SRW Timeout: 166

Learning-Domain: vlan-id 20
Group: 198.51.100.3
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 203.0.113.4, port: ge-1/3/5.20
Downstream port: ge-1/3/3.20
Downstream neighbors:
203.0.113.3 State: Join Flags: SRW Timeout: 168
```

### show pim snooping join instance

```
user@host> show pim snooping join instance vpls1
```

```
Instance: vpls1

Learning-Domain: vlan-id 10
Group: 198.51.100.2
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 192.0.2.4, port: ge-1/3/5.10

Learning-Domain: vlan-id 20
```

```
Group: 198.51.100.3
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 203.0.113.4, port: ge-1/3/5.20
```

### **show pim snooping join vlan-id**

user@host> **show pim snooping join vlan-id 10**

```
Instance: vpls1
Learning-Domain: vlan-id 10
Group: 198.51.100.2
Source: *
Flags: sparse,rptree,wildcard
Upstream state: None
Upstream neighbor: 192.0.2.4, port: ge-1/3/5.10
```

## show pim snooping neighbors

### Syntax

```
show pim snooping neighbors
<brief | detail>
<instance instance-name>
<interface interface-name>
<logical-system logical-system-name>
<vlan-id vlan-identifier>
```

### Release Information

Command introduced in Junos OS Release 12.3 for MX Series 5G Universal Routing Platforms.

Command introduced in Junos OS Release 13.2 for M Series Multiservice Edge devices.

### Description

Display information about Protocol Independent Multicast (PIM) snooping neighbors.

### Options

**none**—Display detailed information.

**brief | detail**—(Optional) Display the specified level of output.

**instance *instance-name***—(Optional) Display PIM snooping neighbor information for the specified routing instance.

**interface *interface-name***—(Optional) Display information for the specified PIM snooping neighbor interface.

**logical-system *logical-system-name***—(Optional) Display information about a particular logical system, or type 'all'.

**vlan-id *vlan-identifier***—(Optional) Display PIM snooping neighbor information for the specified VLAN.

### Required Privilege Level

view

### RELATED DOCUMENTATION

[Configuring Interface Priority for PIM Designated Router Selection](#)

[Modifying the PIM Hello Interval](#)

[PIM Snooping for VPLS](#)

[show pim neighbors](#)



## List of Sample Output

[show pim snooping neighbors on page 330](#)

[show pim snooping neighbors detail on page 330](#)

[show pim snooping neighbors instance on page 332](#)

[show pim snooping neighbors interface on page 333](#)

[show pim snooping neighbors vlan-id on page 333](#)

## Output Fields

[Table 12 on page 329](#) lists the output fields for the **show pim snooping neighbors** command. Output fields are listed in the approximate order in which they appear.

**Table 12: show pim snooping neighbors Output Fields**

Field Name	Field Description	Level of Output
<b>Instance</b>	Routing instance for PIM snooping.	All levels
<b>Learning-Domain</b>	Learning domain for PIM snooping.	All levels
<b>Interface</b>	Router interface for which PIM snooping neighbor details are displayed.	All levels
<b>Option</b>	PIM snooping options available on the specified interface: <ul style="list-style-type: none"> <li>• H = Hello Option Holdtime</li> <li>• P = Hello Option DR Priority</li> <li>• L = Hello Option LAN Prune Delay</li> <li>• G = Generation Identifier</li> <li>• T = Tracking Bit</li> </ul>	All levels
<b>Uptime</b>	Time the neighbor has been operational since the PIM process was last initialized, in the format <b>dd:hh:mm:ss ago</b> for less than a week and <b>nwnd:hh:mm:ss ago</b> for more than a week.	All levels
<b>Neighbor addr</b>	IP address of the PIM snooping neighbor connected through the specified interface.	All levels
<b>Address</b>	IP address of the specified router interface.	All levels
<b>Hello Option Holdtime</b>	Time for which the neighbor is available, in seconds. The range of values is 0 through 65,535.	<b>detail</b>
<b>Hello Option DR Priority</b>	Designated router election priority. The range of values is 0 through 4294967295.  <b>NOTE:</b> By default, every PIM interface has an equal probability (priority 1) of being selected as the DR.	<b>detail</b>

Table 12: show pim snooping neighbors Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Hello Option Generation ID</b>	9-digit or 10-digit number used to tag hello messages.	<b>detail</b>
<b>Hello Option LAN Prune Delay</b>	Time to wait before the neighbor receives prune messages, in the format <b>delay nnn ms override nnnn ms</b> .	<b>detail</b>

## Sample Output

### show pim snooping neighbors

```
user@host> show pim snooping neighbors
```

```
B = Bidirectional Capable, G = Generation Identifier,
H = Hello Option Holdtime, L = Hello Option LAN Prune Delay,
P = Hello Option DR Priority, T = Tracking Bit
```

```
Instance: vpls1
```

```
Learning-Domain: vlan-id 10
```

```
Interface Option Uptime Neighbor addr
```

```
ge-1/3/1.10 HPLGT 00:43:33 192.0.2.2
```

```
ge-1/3/3.10 HPLGT 00:43:33 192.0.2.3
```

```
ge-1/3/5.10 HPLGT 00:43:33 192.0.2.4
```

```
ge-1/3/7.10 HPLGT 00:43:33 192.0.2.5
```

```
Learning-Domain: vlan-id 20
```

```
Interface Option Uptime Neighbor addr
```

```
ge-1/3/1.20 HPLGT 00:43:33 192.0.2.12
```

```
ge-1/3/3.20 HPLGT 00:43:33 192.0.2.13
```

```
ge-1/3/5.20 HPLGT 00:43:33 192.0.2.14
```

```
ge-1/3/7.20 HPLGT 00:43:33 192.0.2.15
```

### show pim snooping neighbors detail

```
user@host> show pim snooping neighbors detail
```

Instance: vpls1  
Learning-Domain: vlan-id 10

Interface: ge-1/3/1.10  
Address: 192.0.2.2  
Uptime: 00:44:51  
Hello Option Holdtime: 105 seconds 83 remaining  
Hello Option DR Priority: 1  
Hello Option Generation ID: 830908833  
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms  
Tracking is supported

Interface: ge-1/3/3.10  
Address: 192.0.2.3  
Uptime: 00:44:51  
Hello Option Holdtime: 105 seconds 97 remaining  
Hello Option DR Priority: 1  
Hello Option Generation ID: 2056520742  
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms  
Tracking is supported

Interface: ge-1/3/5.10  
Address: 192.0.2.4  
Uptime: 00:44:51  
Hello Option Holdtime: 105 seconds 81 remaining  
Hello Option DR Priority: 1  
Hello Option Generation ID: 1152066227  
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms  
Tracking is supported

Interface: ge-1/3/7.10  
Address: 192.0.2.5  
Uptime: 00:44:51  
Hello Option Holdtime: 105 seconds 96 remaining  
Hello Option DR Priority: 1  
Hello Option Generation ID: 1113200338  
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms  
Tracking is supported  
Learning-Domain: vlan-id 20

Interface: ge-1/3/1.20  
Address: 192.0.2.12  
Uptime: 00:44:51  
Hello Option Holdtime: 105 seconds 81 remaining

```

Hello Option DR Priority: 1
Hello Option Generation ID: 963205167
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms
Tracking is supported

```

```

Interface: ge-1/3/3.20
Address: 192.0.2.13
Uptime: 00:44:51
Hello Option Holdtime: 105 seconds 104 remaining
Hello Option DR Priority: 1
Hello Option Generation ID: 166921538
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms
Tracking is supported

```

```

Interface: ge-1/3/5.20
Address: 192.0.2.14
Uptime: 00:44:51
Hello Option Holdtime: 105 seconds 88 remaining
Hello Option DR Priority: 1
Hello Option Generation ID: 789422835
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms
Tracking is supported

```

```

Interface: ge-1/3/7.20
Address: 192.0.2.15
Uptime: 00:44:51
Hello Option Holdtime: 105 seconds 88 remaining
Hello Option DR Priority: 1
Hello Option Generation ID: 1563649680
Hello Option LAN Prune Delay: delay 500 ms override 2000 ms
Tracking is supported

```

### **show pim snooping neighbors instance**

```
user@host> show pim snooping neighbors instance vpls1
```

```

B = Bidirectional Capable, G = Generation Identifier,
H = Hello Option Holdtime, L = Hello Option LAN Prune Delay,
P = Hello Option DR Priority, T = Tracking Bit

```

```

Instance: vpls1
Learning-Domain: vlan-id 10

```

```
Interface Option Uptime Neighbor addr
```

```

ge-1/3/1.10 HPLGT 00:46:03 192.0.2.2
ge-1/3/3.10 HPLGT 00:46:03 192.0.2.3
ge-1/3/5.10 HPLGT 00:46:03 192.0.2.4
ge-1/3/7.10 HPLGT 00:46:03 192.0.2.5

```

```

Learning-Domain: vlan-id 20

```

```

Interface Option Uptime Neighbor addr
ge-1/3/1.20 HPLGT 00:46:03 192.0.2.12
ge-1/3/3.20 HPLGT 00:46:03 192.0.2.13
ge-1/3/5.20 HPLGT 00:46:03 192.0.2.14
ge-1/3/7.20 HPLGT 00:46:03 192.0.2.15

```

### show pim snooping neighbors interface

user@host> show pim snooping neighbors interface ge-1/3/1.20

```

B = Bidirectional Capable, G = Generation Identifier,
H = Hello Option Holdtime, L = Hello Option LAN Prune Delay,
P = Hello Option DR Priority, T = Tracking Bit

```

```

Instance: vpls1
Learning-Domain: vlan-id 10
Learning-Domain: vlan-id 20

```

```

Interface Option Uptime Neighbor addr
ge-1/3/1.20 HPLGT 00:48:04 192.0.2.12

```

### show pim snooping neighbors vlan-id

user@host> show pim snooping neighbors vlan-id 10

```

B = Bidirectional Capable, G = Generation Identifier,
H = Hello Option Holdtime, L = Hello Option LAN Prune Delay,
P = Hello Option DR Priority, T = Tracking Bit

```

```

Instance: vpls1
Learning-Domain: vlan-id 10

```

```

Interface Option Uptime Neighbor addr
ge-1/3/1.10 HPLGT 00:49:12 192.0.2.2
ge-1/3/3.10 HPLGT 00:49:12 192.0.2.3

```

```
ge-1/3/5.10 HPLGT 00:49:12 192.0.2.4  
ge-1/3/7.10 HPLGT 00:49:12 192.0.2.5
```

## show pim snooping statistics

### Syntax

```
show pim snooping statistics
<instance instance-name>
<interface interface-name>
<logical-system logical-system-name>
<vlan-id vlan-id>
```

### Release Information

Command introduced in Junos OS Release 12.3 for MX Series 5G Universal Routing Platforms.

Command introduced in Junos OS Release 13.2 for M Series Multiservice Edge devices.

### Description

Display Protocol Independent Multicast (PIM) snooping statistics.

### Options

**none**—Display PIM statistics.

**instance *instance-name***—(Optional) Display statistics for a specific routing instance enabled by Protocol Independent Multicast (PIM) snooping.

**interface *interface-name***—(Optional) Display statistics about the specified interface for PIM snooping.

**logical-system *logical-system-name***—(Optional) Display information about a particular logical system, or type 'all'.

**vlan-id *vlan-identifier***—(Optional) Display PIM snooping statistics information for the specified VLAN.

### Required Privilege Level

view

### RELATED DOCUMENTATION

*PIM Snooping for VPLS*

[clear pim snooping statistics](#) | 284

### List of Sample Output

[show pim snooping statistics on page 337](#)

[show pim snooping statistics instance on page 338](#)

[show pim snooping statistics interface on page 339](#)

[show pim snooping statistics vlan-id on page 340](#)

## Output Fields

Table 13 on page 336 lists the output fields for the **show pim snooping statistics** command. Output fields are listed in the approximate order in which they appear.

Table 13: show pim snooping statistics Output Fields

Field Name	Field Description	Level of Output
<b>Instance</b>	Routing instance for PIM snooping.	All levels
<b>Learning-Domain</b>	Learning domain for PIM snooping.	All levels
<b>Tx J/P messages</b>	Total number of transmitted join/prune packets.	All levels
<b>RX J/P messages</b>	Total number of received join/prune packets.	All levels
<b>Rx J/P messages -- seen</b>	Number of join/prune packets seen but not received on the upstream interface.	All levels
<b>Rx J/P messages -- received</b>	Number of join/prune packets received on the downstream interface.	All levels
<b>Rx Hello messages</b>	Total number of received hello packets.	All levels
<b>Rx Version Unknown</b>	Number of packets received with an unknown version number.	All levels
<b>Rx Neighbor Unknown</b>	Number of packets received from an unknown neighbor.	All levels
<b>Rx Upstream Neighbor Unknown</b>	Number of packets received with unknown upstream neighbor information.	All levels
<b>Rx Bad Length</b>	Number of packets received containing incorrect length information.	All levels
<b>Rx J/P Busy Drop</b>	Number of join/prune packets dropped while the router is busy.	All levels
<b>Rx J/P Group Aggregate 0</b>	Number of join/prune packets received containing the aggregate group information.	All levels
<b>Rx Malformed Packet</b>	Number of malformed packets received.	All levels



Table 13: show pim snooping statistics Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Rx No PIM Interface</b>	Number of packets received without the interface information.	All levels
<b>Rx No Upstream Neighbor</b>	Number of packets received without upstream neighbor information.	All levels
<b>Rx Unknown Hello Option</b>	Number of hello packets received with unknown options.	All levels

## Sample Output

**show pim snooping statistics**

user@host> **show pim snooping statistics**

```

Instance: vpls1
Learning-Domain: vlan-id 10

Tx J/P messages 0
RX J/P messages 8
Rx J/P messages -- seen 0
Rx J/P messages -- received 8
Rx Hello messages 37
Rx Version Unknown 0
Rx Neighbor Unknown 0
Rx Upstream Neighbor Unknown 0
Rx Bad Length 0
Rx J/P Busy Drop 0
Rx J/P Group Aggregate 0
Rx Malformed Packet 0
Rx No PIM Interface 0
Rx No Upstream Neighbor 0
Rx Bad Length 0
Rx Neighbor Unknown 0
Rx Unknown Hello Option 0
Rx Malformed Packet 0

Learning-Domain: vlan-id 20

```

```

Tx J/P messages 0
RX J/P messages 2
Rx J/P messages -- seen 0
Rx J/P messages -- received 2
Rx Hello messages 39
Rx Version Unknown 0
Rx Neighbor Unknown 0
Rx Upstream Neighbor Unknown 0
Rx Bad Length 0
Rx J/P Busy Drop 0
Rx J/P Group Aggregate 0
Rx Malformed Packet 0
Rx No PIM Interface 0
Rx No Upstream Neighbor 0
Rx Bad Length 0
Rx Neighbor Unknown 0
Rx Unknown Hello Option 0
Rx Malformed Packet 0

```

### **show pim snooping statistics instance**

user@host> **show pim snooping statistics instance vpls1**

```

Instance: vpls1
Learning-Domain: vlan-id 10

Tx J/P messages 0
RX J/P messages 9
Rx J/P messages -- seen 0
Rx J/P messages -- received 9
Rx Hello messages 45
Rx Version Unknown 0
Rx Neighbor Unknown 0
Rx Upstream Neighbor Unknown 0
Rx Bad Length 0
Rx J/P Busy Drop 0
Rx J/P Group Aggregate 0
Rx Malformed Packet 0
Rx No PIM Interface 0
Rx No Upstream Neighbor 0
Rx Bad Length 0
Rx Neighbor Unknown 0
Rx Unknown Hello Option 0

```

```
Rx Malformed Packet 0
```

```
Learning-Domain: vlan-id 20
```

```
Tx J/P messages 0
```

```
RX J/P messages 3
```

```
Rx J/P messages -- seen 0
```

```
Rx J/P messages -- received 3
```

```
Rx Hello messages 47
```

```
Rx Version Unknown 0
```

```
Rx Neighbor Unknown 0
```

```
Rx Upstream Neighbor Unknown 0
```

```
Rx Bad Length 0
```

```
Rx J/P Busy Drop 0
```

```
Rx J/P Group Aggregate 0
```

```
Rx Malformed Packet 0
```

```
Rx No PIM Interface 0
```

```
Rx No Upstream Neighbor 0
```

```
Rx Bad Length 0
```

```
Rx Neighbor Unknown 0
```

```
Rx Unknown Hello Option 0
```

```
Rx Malformed Packet 0
```

### **show pim snooping statistics interface**

```
user@host> show pim snooping statistics interface ge-1/3/1.20
```

```
Instance: vpls1
```

```
Learning-Domain: vlan-id 10
```

```
Learning-Domain: vlan-id 20
```

```
PIM Interface statistics for ge-1/3/1.20
```

```
Tx J/P messages 0
```

```
RX J/P messages 0
```

```
Rx J/P messages -- seen 0
```

```
Rx J/P messages -- received 0
```

```
Rx Hello messages 13
```

```
Rx Version Unknown 0
```

```
Rx Neighbor Unknown 0
```

```
Rx Upstream Neighbor Unknown 0
```

```
Rx Bad Length 0
```

```
Rx J/P Busy Drop 0
```

```
Rx J/P Group Aggregate 0  
Rx Malformed Packet 0
```

### **show pim snooping statistics vlan-id**

user@host> **show pim snooping statistics vlan-id 10**

```
Instance: vpls1  
Learning-Domain: vlan-id 10  
  
Tx J/P messages 0  
RX J/P messages 11  
Rx J/P messages -- seen 0  
Rx J/P messages -- received 11  
Rx Hello messages 64  
Rx Version Unknown 0  
Rx Neighbor Unknown 0  
Rx Upstream Neighbor Unknown 0  
Rx Bad Length 0  
Rx J/P Busy Drop 0  
Rx J/P Group Aggregate 0  
Rx Malformed Packet 0  
Rx No PIM Interface 0  
Rx No Upstream Neighbor 0  
Rx Bad Length 0  
Rx Neighbor Unknown 0
```

## show route

### List of Syntax

[Syntax on page 341](#)

[Syntax \(EX Series Switches\) on page 341](#)

### Syntax

```
show route
<all>
<destination-prefix>
<logical-system (all | logical-system-name)>
<private>
<te-ipv4-prefix-ip te-ipv4-prefix-ip>
<te-ipv4-prefix-node-ip te-ipv4-prefix-node-ip>
<te-ipv4-prefix-node-iso te-ipv4-prefix-node-iso>
<rib-sharding (main | rib-shard-name)>
```

### Syntax (EX Series Switches)

```
show route
<all>
<destination-prefix>
<private>
```

### Release Information

Command introduced before Junos OS Release 7.4.

Command introduced in Junos OS Release 9.0 for EX Series switches.

Option **private** introduced in Junos OS Release 9.5.

Option **private** introduced in Junos OS Release 9.5 for EX Series switches.

Command introduced in Junos OS Release 15.1R3 on MX Series routers for enhanced subscriber management.

Option **display-client-data** introduced in Junos OS Release 16.2R1 on MX80, MX104, MX240, MX480, MX960, MX2010, MX2020, vMX Series routers.

Options **te-ipv4-prefix-ip**, **te-ipv4-prefix-node-ip**, and **te-ipv4-prefix-node-iso** introduced in Junos OS Release 17.2R1 on MX Series and PTX Series.

**rib-sharding** option introduced in cRPD Release 20.1R1.

### Description

Display the active entries in the routing tables.

### Options

**none**—Display brief information about all active entries in the routing tables.

**all**—(Optional) Display information about all routing tables, including private, or internal, routing tables.

**destination-prefix**—(Optional) Display active entries for the specified address or range of addresses.

**logical-system (all | *logical-system-name*)**—(Optional) Perform this operation on all logical systems or on a particular logical system.

**private**—(Optional) Display information only about all private, or internal, routing tables.

**display-client-data** —(Optional) Display client id and cookie information for routes installed by the routing protocol process client applications.

**te-ipv4-prefix-ip *te-ipv4-prefix-ip***—(Optional) Display IPv4 address of the traffic-engineering prefix, without the mask length if present in the routing table.

**te-ipv4-prefix-node-ip *te-ipv4-prefix-node-ip***—(Optional) Display all prefixes that have originated from the traffic-engineering node. You can filter IPv4 node addresses from the traffic-engineered routes in the **lsdist.0** table.

**te-ipv4-prefix-node-iso *te-ipv4-prefix-node-iso***—(Optional) Display all prefixes that have originated from the traffic-engineering node. You can filter IPv4 routes with the specified ISO circuit ID from the **lsdist.0** table.

**rib-sharding (main | *rib-shard-name*)**—(Optional) Display the rib shard name.

### Required Privilege Level

view

### RELATED DOCUMENTATION

*Understanding IS-IS Configuration*

*Example: Configuring IS-IS*

*Examples: Configuring Internal BGP Peering*

*Examples: Configuring External BGP Peering*

*Examples: Configuring OSPF Routing Policy*

*Verifying and Managing Junos OS Enhanced Subscriber Management*

### List of Sample Output

[show route on page 346](#)

[show route \(VPN\) on page 347](#)

[show route \(with Destination Prefix\) on page 348](#)

[show route destination-prefix detail on page 348](#)

[show route extensive on page 348](#)

[show route extensive \( ECMP\) on page 349](#)

[show route extensive \(Multipath Resolution\) on page 349](#)  
[show route \(Enhanced Subscriber Management\) on page 355](#)  
[show route \(IPv6 Flow Specification\) on page 356](#)  
[show route display-client-data detail on page 356](#)  
[show route rib-sharding on page 357](#)  
[show route te-ipv4-prefix-ip on page 358](#)  
[show route te-ipv4-prefix-ip extensive on page 359](#)  
[show route te-ipv4-prefix-node-iso on page 362](#)  
[show route te-ipv4-prefix-node-iso extensive on page 363](#)  
[show route te-ipv4-prefix-node-iso detail on page 366](#)  
[show route rib-sharding junos-bgpshard14 on page 370](#)

## Output Fields

Table 14 on page 343 describes the output fields for the **show route** command. Output fields are listed in the approximate order in which they appear.

Table 14: show route Output Fields

Field Name	Field Description
<i>routing-table-name</i>	Name of the routing table (for example, inet.0).
<i>number destinations</i>	Number of destinations for which there are routes in the routing table.
<i>number routes</i>	<p>Number of routes in the routing table and total number of routes in the following states:</p> <ul style="list-style-type: none"> <li>• <b>active</b> (routes that are active).</li> <li>• <b>holddown</b> (routes that are in the pending state before being declared inactive). A holddown route was once the active route and is no longer the active route. The route is in the holddown state because a protocol still has interest in the route, meaning that the interest bit is set. A protocol might have its interest bit set on the previously active route because the protocol is still advertising the route. The route will be deleted after all protocols withdraw their advertisement of the route and remove their interest bit. A persistent holddown state often means that the interested protocol is not releasing its interest bit properly.</li> </ul> <p>However, if you have configured advertisement of multiple routes (with the <b>add-path</b> or <b>advertise-inactive</b> statement), the holddown bit is most likely set because BGP is advertising the route as an active route. In this case, you can ignore the holddown state because nothing is wrong.</p> <p>If you have configured <b>uRPF-loose</b> mode, the holddown bit is most likely set because Kernel Routing Table (KRT) is using inactive route to build valid incoming interfaces. In this case, you can ignore the holddown state because nothing is wrong.</p> <ul style="list-style-type: none"> <li>• <b>hidden</b> (routes that are not used because of a routing policy).</li> </ul>

Table 14: show route Output Fields (*continued*)

Field Name	Field Description
<i>destination-prefix</i>	<p>Route destination (for example:10.0.0.1/24). Sometimes the route information is presented in another format, such as:</p> <ul style="list-style-type: none"> <li>• <b>MPLS-label</b> (for example, 80001).</li> <li>• <b>interface-name</b> (for example, ge-1/0/2).</li> <li>• <b>neighbor-address:control-word-status:encapsulation type:vc-id :source</b> (Layer 2 circuit only. For example, 10.1.1.195:NoCtrlWord:1:1:Local/96): <ul style="list-style-type: none"> <li>• <b>neighbor-address</b>—Address of the neighbor.</li> <li>• <b>control-word-status</b>—Whether the use of the control word has been negotiated for this virtual circuit: <b>NoCtrlWord</b> or <b>CtrlWord</b>.</li> <li>• <b>encapsulation type</b>—Type of encapsulation, represented by a number: (1) Frame Relay DLCI, (2) ATM AAL5 VCC transport, (3) ATM transparent cell transport, (4) Ethernet, (5) VLAN Ethernet, (6) HDLC, (7) PPP, (8) ATM VCC cell transport, (10) ATM VPC cell transport.</li> <li>• <b>vc-id</b>—Virtual circuit identifier.</li> <li>• <b>source</b>—Source of the advertisement: <b>Local</b> or <b>Remote</b>.</li> </ul> </li> </ul>
[ <i>protocol, preference</i> ]	<p>Protocol from which the route was learned and the preference value for the route.</p> <ul style="list-style-type: none"> <li>• <b>+</b>—A plus sign indicates the active route, which is the route installed from the routing table into the forwarding table.</li> <li>• <b>-</b>—A hyphen indicates the last active route.</li> <li>• <b>*</b>—An asterisk indicates that the route is both the active and the last active route. An asterisk before a <b>to</b> line indicates the best subpath to the route.</li> </ul> <p>In every routing metric except for the BGP <b>LocalPref</b> attribute, a lesser value is preferred. In order to use common comparison routines, Junos OS stores the 1's complement of the <b>LocalPref</b> value in the <b>Preference2</b> field. For example, if the <b>LocalPref</b> value for Route 1 is 100, the <b>Preference2</b> value is -101. If the <b>LocalPref</b> value for Route 2 is 155, the <b>Preference2</b> value is -156. Route 2 is preferred because it has a higher <b>LocalPref</b> value and a lower <b>Preference2</b> value.</p>
<i>weeks:days</i> <i>hours:minutes:seconds</i>	How long the route been known (for example, <b>2w4d 13:11:14</b> , or 2 weeks, 4 days, 13 hours, 11 minutes, and 14 seconds).
<b>metric</b>	Cost value of the indicated route. For routes within an AS, the cost is determined by the IGP and the individual protocol metrics. For external routes, destinations, or routing domains, the cost is determined by a preference value.
<b>localpref</b>	Local preference value included in the route.



Table 14: show route Output Fields (*continued*)

Field Name	Field Description
<b>from</b>	Interface from which the route was received.
<b>AS path</b>	<p>AS path through which the route was learned. The letters at the end of the AS path indicate the path origin, providing an indication of the state of the route at the point at which the AS path originated:</p> <ul style="list-style-type: none"> <li>• <b>I</b>—IGP.</li> <li>• <b>E</b>—EGP.</li> <li>• <b>?</b>—Incomplete; typically, the AS path was aggregated.</li> </ul> <p>When AS path numbers are included in the route, the format is as follows:</p> <ul style="list-style-type: none"> <li>• <b>[ ]</b>—Brackets enclose the local AS number associated with the AS path if more than one AS number is configured on the routing device, or if AS path prepending is configured.</li> <li>• <b>{ }</b>—Braces enclose AS sets, which are groups of AS numbers in which the order does not matter. A set commonly results from route aggregation. The numbers in each AS set are displayed in ascending order.</li> <li>• <b>( )</b>—Parentheses enclose a confederation.</li> <li>• <b>( [ ] )</b>—Parentheses and brackets enclose a confederation set.</li> </ul> <p><b>NOTE:</b> In Junos OS Release 10.3 and later, the AS path field displays an unrecognized attribute and associated hexadecimal value if BGP receives attribute 128 (attribute set) and you have not configured an independent domain in any routing instance.</p>
<b>encapsulated</b>	Extended next-hop encoding capability enabled for the specified BGP community for routing IPv4 traffic over IPv6 tunnels. When BGP receives routes without the tunnel community, IPv4-Over IPv6 tunnels are not created and BGP routes are resolved without encapsulation.
<b>Route Labels</b>	Stack of labels carried in the BGP route update.
<b>validation-state</b>	<p>(BGP-learned routes) Validation status of the route:</p> <ul style="list-style-type: none"> <li>• <b>Invalid</b>—Indicates that the prefix is found, but either the corresponding AS received from the EBGp peer is not the AS that appears in the database, or the prefix length in the BGP update message is longer than the maximum length permitted in the database.</li> <li>• <b>Unknown</b>—Indicates that the prefix is not among the prefixes or prefix ranges in the database.</li> <li>• <b>Unverified</b>—Indicates that the origin of the prefix is not verified against the database. This is because the database got populated and the validation is not called for in the BGP import policy, although origin validation is enabled, or the origin validation is not enabled for the BGP peers.</li> <li>• <b>Valid</b>—Indicates that the prefix and autonomous system pair are found in the database.</li> </ul>

Table 14: show route Output Fields (*continued*)

Field Name	Field Description
<b>to</b>	<p>Next hop to the destination. An angle bracket (&gt;) indicates that the route is the selected route.</p> <p>If the destination is <b>Discard</b>, traffic is dropped.</p>
<b>via</b>	<p>Interface used to reach the next hop. If there is more than one interface available to the next hop, the interface that is actually used is followed by the word <b>Selected</b>. This field can also contain the following information:</p> <ul style="list-style-type: none"> <li>• <b>Weight</b>—Value used to distinguish primary, secondary, and fast reroute backup routes. Weight information is available when MPLS label-switched path (LSP) link protection, node-link protection, or fast reroute is enabled, or when the standby state is enabled for secondary paths. A lower weight value is preferred. Among routes with the same weight value, load balancing is possible.</li> <li>• <b>Balance</b>—Balance coefficient indicating how traffic of unequal cost is distributed among next hops when a routing device is performing unequal-cost load balancing. This information is available when you enable BGP multipath load balancing.</li> <li>• <b>lsp-path-name</b>—Name of the LSP used to reach the next hop.</li> <li>• <b>label-action</b>—MPLS label and operation occurring at the next hop. The operation can be <b>pop</b> (where a label is removed from the top of the stack), <b>push</b> (where another label is added to the label stack), or <b>swap</b> (where a label is replaced by another label). For VPNs, expect to see multiple <b>push</b> operations, corresponding to the inner and outer labels required for VPN routes (in the case of a direct PE-to-PE connection, the VPN route would have the inner label push only).</li> </ul>
<b>Private unicast</b>	(Enhanced subscriber management for MX Series routers) Indicates that an access-internal route is managed by enhanced subscriber management. By contrast, access-internal routes <i>not</i> managed by enhanced subscriber management are displayed with associated next-hop and media access control (MAC) address information.
balance	Distribution of the load based on the underlying operational interface bandwidth for equal-cost multipaths (ECMP) across the nexthop gateways in percentages.

## Sample Output

```
show route
```

```
user@host> show route
```

```

inet.0: 11 destinations, 12 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1:65500:1:10.0.0.20/240
    *[MVPN/70] 19:53:41, metric2 1
        Indirect
1:65500:1:10.0.0.40/240
    *[BGP/170] 19:53:29, localpref 100, from 10.0.0.30
        AS path: I
        > to 10.0.24.4 via lt-0/3/0.24, label-switched-path toD
    [BGP/170] 19:53:26, localpref 100, from 10.0.0.33
        AS path: I
        > to 10.0.24.4 via lt-0/3/0.24, label-switched-path toD
1:65500:1:10.0.0.60/240
    *[BGP/170] 19:53:29, localpref 100, from 10.0.0.30
        AS path: I
        > to 10.0.28.8 via lt-0/3/0.28, label-switched-path toF
    [BGP/170] 19:53:25, localpref 100, from 10.0.0.33
        AS path: I
        > to 10.0.28.8 via lt-0/3/0.28, label-switched-path toF

```

### show route (VPN)

The following sample output shows a VPN route with composite next hops enabled. The first **Push** operation corresponds to the outer label. The second **Push** operation corresponds to the inner label.

user@host> **show route 192.0.2.0**

```

13979:66500l.inet.0: 871 destinations, 3556 routes (871 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both

192.0.2.0/24    [BGP/170] 00:28:32, localpref 100, from 10.9.9.160
                AS path: 13980 ?, validation-state: unverified
                > to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
    [BGP/170] 00:28:28, localpref 100, from 10.9.9.169
                AS path: 13980 ?, validation-state: unverified
                > to 10.100.0.42 via ae2.0, Push 126016, Push 300368(top)
#[Multipath/255] 00:28:28, metric2 102
                > to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)
                to 10.100.0.42 via ae2.0, Push 16, Push 300368(top)

```

**show route (with Destination Prefix)**

```
user@host> show route 192.168.0.0/12
```

```
inet.0: 10 destinations, 10 routes (9 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.0.0/12      *[Static/5] 2w4d 12:54:27
                    > to 192.168.167.254 via fxp0.0
```

**show route destination-prefix detail**

```
user@host> show route 198.51.100.0 detail
```

```
inet.0: 15 destinations, 20 routes (15 active, 0 holddown, 0 hidden)
198.51.100.0/24 (2 entries, 2 announced)
    *BGP      Preference: 170/-101
    ...
    BGP-Static Preference: 4294967292
        Next hop type: Discard
        Address: 0x9041ae4
        Next-hop reference count: 2
        State: <NoReadvrt Int Ext AlwaysFlash>
    Inactive reason: Route Preference
    Local AS: 200
    Age: 4d 1:40:40
    Validation State: unverified
    Task: RT
    Announcement bits (1): 2-BGP_RT_Background
    AS path: 4 5 6 I
```

**show route extensive**

```
user@host> show route extensive
```

```
vl.mvpn.0: 5 destinations, 8 routes (5 active, 1 holddown, 0 hidden)
1:65500:1:10.0.0.40/240 (1 entry, 1 announced)
    *BGP      Preference: 170/-101
    PMSI: Flags 0x0: Label[0:0:0]: PIM-SM: Sender 10.0.0.40 Group 203.0.113.1

    Next hop type: Indirect
    Address: 0x92455b8
```

```

Next-hop reference count: 2
Source: 10.0.0.30
Protocol next hop: 10.0.0.40
Indirect next hop: 2 no-forward
State: <Active Int Ext>
      Local AS: 64510 Peer AS: 64511
Age: 3 Metric2: 1
Validation State: unverified
Task: BGP_64510.10.0.0.30+179
Announcement bits (2): 0-PIM.v1 1-mvpn global task
AS path: I (Originator) Cluster list: 10.0.0.30
AS path: Originator ID: 10.0.0.40
Communities: target:64502:100 encapsulation:0L:14 Import
Accepted
Localpref: 100
Router ID: 10.0.0.30
Primary Routing Table bgp.mvpn.0
Indirect next hops: 1
  Protocol next hop: 10.0.0.40 Metric: 1
  Indirect next hop: 2 no-forward
  Indirect path forwarding next hops: 1
    Next hop type: Router
    Next hop: 10.0.24.4 via lt-0/3/0.24 weight 0x1
    10.0.0.40/32 Originating RIB: inet.3
    Metric: 1 Node path count: 1
    Forwarding nexthops: 1
      Nexthop: 10.0.24.4 via lt-0/3/0.24

```

### show route extensive ( ECMP)

user@host> show route extensive

```

*IS-IS Preference: 15
  Level: 1
  Next hop type: Router, Next hop index: 1048577
  Address: 0xFFFFFFFF
  Next-hop reference count: YY
  Next hop: 198.51.100.2 via ae1.0 balance 43%, selected
  Session Id: 0x141
  Next hop: 192.0.2.2 via ae0.0 balance 57%

```

### show route extensive (Multipath Resolution)

user@host> show route extensive

```

inet.0: 37 destinations, 37 routes (36 active, 0 holddown, 1 hidden)
10.1.1.2/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 10.1.1.2/32 -> {indirect(1048574)}
  *Static Preference: 5
    Next hop type: Indirect, Next hop index: 0
    Address: 0xb39dlb0
    Next-hop reference count: 2
    Next hop type: Router, Next hop index: 581
    Next hop: 10.1.1.2 via ge-2/0/1.0, selected
    Session Id: 0x144
    Next hop: 10.2.1.2 via ge-2/0/2.0, selected
    Session Id: 0x145
    Protocol next hop: 10.1.1.1
    Indirect next hop: 0xb2b20f0 1048574 INH Session ID: 0x143
    State: <Active Int Ext>
    Age: 2:53 Metric2: 0
    Validation State: unverified
    Task: RT
    Announcement bits (2): 0-KRT 2-Resolve tree 1
    AS path: I
    Indirect next hops: 1
      Protocol next hop: 10.1.1.1
      Indirect next hop: 0xb2b20f0 1048574 INH Session ID: 0x143

      Indirect path forwarding next hops: 2
        Next hop type: Router
        Next hop: 10.1.1.2 via ge-2/0/1.0
        Session Id: 0x144
        Next hop: 10.2.1.2 via ge-2/0/2.0
        Session Id: 0x145
    10.1.1.1/32 Originating RIB: inet.0
    Node path count: 1
    Node flags: 1
    Forwarding nexthops: 2 (Merged)
    Nexthop: 10.1.1.2 via ge-2/0/1.0
    Nexthop: 10.2.1.2 via ge-2/0/2.0

```

```
user@host> show route active-path extensive
```

```
user@host> show route 198.51.100.1 active-path extensive
```

```

inet.0: 1000061 destinations, 1000082 routes (1000061 active, 0 holddown, 0 hidden)
198.51.100.1/32 (1 entry, 1 announced)

```

```

TSI:
KRT in-kernel 198.51.100.1/32 -> {indirect(1051215)}
unicast reverse-path: 0
[ae0.0 ae1.0]
Page 0 idx 0, (group Internet-IPv4 type External) Type 1 val 0xbb2e53d8 (adv_entry)
Advertised metrics:
Nexthop: Self
AS path: [500] 410 I
Communities:
Path 198.51.100.1 from 10.0.0.11 Vector len 4. Val: 0
*BGP Preference: 170/-101
Next hop type: Indirect, Next hop index: 0
Address: 0x2e9aacdc
Next-hop reference count: 500000
Source: 10.0.0.11
Next hop type: Router, Next hop index: 0
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Label operation: Push 3851, Push 25, Push 20(top)
Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 25: None; Label 20: None;
Label element ptr: 0xb5dc1780
Label parent element ptr: 0x18d48080
Label element references: 2
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Label operation: Push 3851, Push 25, Push 22(top)
Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 25: None; Label 22: None;
Label element ptr: 0xb5dc1700
Label parent element ptr: 0x18d41000
Label element references: 2
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Label operation: Push 3851, Push 24, Push 48(top)
Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 24: None; Label 48: None;
Label element ptr: 0x18d40800
Label parent element ptr: 0x18d49780
Label element references: 3
Label element child references: 0

```

```

Label element lsp id: 0
Session Id: 0x0
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Label operation: Push 3851, Push 24, Push 49(top)
Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 24: None; Label 49: None;
Label element ptr: 0xb5dc1680
Label parent element ptr: 0x18d48f00
Label element references: 2
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1
Label operation: Push 3851, Push 25, Push 21(top)
Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 25: None; Label 21: None;
Label element ptr: 0xb5dc1600
Label parent element ptr: 0x18d44d80
Label element references: 2
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1
Label operation: Push 3851, Push 25, Push 25(top)
Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 25: None; Label 25: None;
Label element ptr: 0xb5dc1580
Label parent element ptr: 0x18d3da80
Label element references: 2
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1, selected
Label operation: Push 3851, Push 24, Push 68(top)
Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 24: None; Label 68: None;
Label element ptr: 0x18d41500
Label parent element ptr: 0x18d49000
Label element references: 3
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1
Label operation: Push 3851, Push 24, Push 69(top)

```



```

Label TTL action: prop-ttl, prop-ttl, prop-ttl(top)
Load balance label: Label 3851: None; Label 24: None; Label 69: None;
Label element ptr: 0xb5dc1500
Label parent element ptr: 0x18d48300
Label element references: 2
Label element child references: 0
Label element lsp id: 0
Session Id: 0x0
Protocol next hop: 10.0.0.11
Label operation: Push 3851
Label TTL action: prop-ttl
Load balance label: Label 3851: None;
Indirect next hop: 0x1883e200 1051215 INH Session ID: 0xb0d
State:
Local AS: 500 Peer AS: 500
Age: 1:40:03 Metric2: 2
Validation State: unverified
Task: BGP_500.10.0.0.11
Announcement bits (5): 0-KRT 8-KRT 9-BGP_RT_Background 10-Resolve tree 5 11-Resolve
    tree 8
AS path: 410 I
Accepted
Route Label: 3851
Localpref: 100
Router ID: 10.0.0.11
Indirect next hops: 1
Protocol next hop: 10.0.0.11 Metric: 2
Label operation: Push 3851
Label TTL action: prop-ttl
Load balance label: Label 3851: None;
Indirect next hop: 0x1883e200 1051215 INH Session ID: 0xb0d
Indirect path forwarding next hops (Merged): 8
Next hop type: Router
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Session Id: 0x0
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Session Id: 0x0
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Session Id: 0x0
Next hop: 10.0.12.2 via ae0.0 weight 0x1
Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1
Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1

```

```

Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1
Session Id: 0x0
Next hop: 10.0.13.3 via ae1.0 weight 0x1
Session Id: 0x0
10.0.0.11/32 Originating RIB: inet.3
Metric: 1 Node path count: 4
Node flags: 1
Indirect nexthops: 4
Protocol Nexthop: 10.0.0.4 Metric: 1 Push 24
Indirect nexthop: 0x1880f200 1048597 INH Session ID: 0xb0c
Path forwarding nexthops link: 0x36120400
Path inh link: 0x0
Indirect path forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0
10.0.0.4/32 Originating RIB: inet.3
Metric: 1 Node path count: 1
Forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0
Protocol Nexthop: 10.0.0.5 Metric: 1 Push 24
Indirect nexthop: 0x18810000 1048596 INH Session ID: 0xb0b
Path forwarding nexthops link: 0x1545be00
Path inh link: 0x0
Indirect path forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0
10.0.0.5/32 Originating RIB: inet.3
Metric: 1 Node path count: 1
Forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0
Protocol Nexthop: 10.0.0.6 Metric: 1 Push 25
Indirect nexthop: 0x1880e600 1048588 INH Session ID: 0xb0a
Path forwarding nexthops link: 0x3611f440

```

```

Path inh link: 0x0
Indirect path forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0
10.0.0.6/32 Originating RIB: inet.3
Metric: 1 Node path count: 1
Forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0
Protocol Nexthop: 10.0.0.7 Metric: 1 Push 25
Indirect nexthop: 0xl880dc00 1048586 INH Session ID: 0xb09
Path forwarding nexthops link: 0x15466d80
Path inh link: 0x0
Indirect path forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0
10.0.0.7/32 Originating RIB: inet.3
Metric: 1 Node path count: 1
Forwarding nexthops: 2
Nexthop: 10.0.12.2 via ae0.0
Session Id: 0
Nexthop: 10.0.13.3 via ae1.0
Session Id: 0

```

### show route (Enhanced Subscriber Management)

user@host> show route

```

inet.0: 41 destinations, 41 routes (40 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

198.51.100.11/24    *[Access-internal/12] 00:00:08
                  > to #0 10.0.0.1.93.65 via demux0.1073741824
198.51.100.12/24    *[Access-internal/12] 00:00:08
                  Private unicast

```

**show route (IPv6 Flow Specification)**

```
user@host> show route
```

```
inet6.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::10:255:185:19/128
    *[Direct/0] 05:11:27
    > via lo0.0
2001:db8::11:11:11:0/120
    *[BGP/170] 00:28:58, localpref 100
    AS path: 2000 I, validation-state: unverified
    > to 2001:db8::13:14:2:2 via ge-1/1/4.0
2001:db8::13:14:2:0/120*[Direct/0] 00:45:07
    > via ge-1/1/4.0
2001:db8::13:14:2:1/128*[Local/0] 00:45:18
    Local via ge-1/1/4.0
fe80::2a0:a50f:fc71:71d5/128
    *[Direct/0] 05:11:27
    > via lo0.0
fe80::5e5e:abff:feb0:933e/128
    *[Local/0] 00:45:18
    Local via ge-1/1/4.0

inet6flow.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::11:11:11:10/128,*,proto=6,dstport=80,srcport=65535/term:1
    *[BGP/170] 00:28:58, localpref 100, from 2001:db8::13:14:2:2
    AS path: 2000 I, validation-state: unverified
    Fictitious
2001:db8::11:11:11:30/128,*,icmp6-type=128,len=100,dscp=10/term:2
    *[BGP/170] 00:20:54, localpref 100, from 2001:db8::13:14:2:2
    AS path: 2000 I, validation-state: unverified
    Fictitious
```

**show route display-client-data detail**

```
user@host> show route 198.51.100.0/24 display-client-data detail
```

```
inet.0: 59 destinations, 70 routes (59 active, 0 holddown, 0 hidden)
198.51.100.0/24 (1 entry, 1 announced)
    State: <FlashAll>
```

```

*BGP-Static Preference: 5/-101
  Next hop type: Indirect, Next hop index: 0
  Address: 0xa5c2af8
  Next-hop reference count: 2
  Next hop type: Router, Next hop index: 1641
  Next hop: 192.0.2.1 via ge-2/1/1.0, selected
  Session Id: 0x160
  Protocol next hop: 192.0.2.1
  Indirect next hop: 0xa732cb0 1048621 INH Session ID: 0x17e
  State: <Active Int Ext AlwaysFlash NSR-incapable Programmed>
  Age: 3:13      Metric2: 0
  Validation State: unverified
  Announcement bits (3): 0-KRT 5-LDP 6-Resolve tree 3
  AS path: I
  Client id: 1, Cookie: 1

```

### show route rib-sharding

user@host> **show route rib-sharding shard-name**

```

inet.0: 193295 destinations, 386345 routes (193295 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1.0.4.0/22      *[BGP/170] 03:35:38, localpref 100, from 1.1.1.1
                AS path: 69 10458 14203 4826 38803 56203 I, validation-state:
                unverified
                > to 10.205.191.254 via fxp0.0
                [BGP/170] 03:35:38, localpref 100, from 1.1.1.2
                AS path: 69 10458 14203 4826 38803 56203 I, validation-state:
                unverified
                > to 10.205.191.254 via fxp0.0
1.0.6.0/24      *[BGP/170] 03:35:38, localpref 100, from 1.1.1.1
                AS path: 69 10458 14203 4826 38803 56203 I, validation-state:
                unverified
                > to 10.205.191.254 via fxp0.0
                [BGP/170] 03:35:38, localpref 100, from 1.1.1.2
                AS path: 69 10458 14203 4826 38803 56203 I, validation-state:
                unverified
                > to 10.205.191.254 via fxp0.0
1.0.64.0/18     *[BGP/170] 03:35:38, localpref 100, from 1.1.1.1
                AS path: 69 10458 14203 2914 2497 7670 18144 I,
validation-state: unverified
                > to 10.205.191.254 via fxp0.0
                [BGP/170] 03:35:38, localpref 100, from 1.1.1.2

```

```

AS path: 69 10458 14203 2914 2497 7670 18144 I,
validation-state: unverified
> to 10.205.191.254 via fxp0.0

```

### show route te-ipv4-prefix-ip

user@host> show route te-ipv4-prefix-ip 10.10.10.10

```

lsdist.0: 283 destinations, 283 routes (283 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L1:0
}/1152
      *[IS-IS/15] 00:01:01
      Fictitious
PREFIX { Node { AS:64496 ISO:0100.0101.0101.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
      *[IS-IS/18] 00:01:01
      Fictitious
PREFIX { Node { AS:64496 ISO:0100.0202.0202.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
      *[IS-IS/18] 00:01:01
      Fictitious
PREFIX { Node { AS:64496 ISO:0100.0303.0303.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
      *[IS-IS/18] 00:01:01
      Fictitious
PREFIX { Node { AS:64496 ISO:0100.0404.0404.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
      *[IS-IS/18] 00:01:01
      Fictitious
PREFIX { Node { AS:64496 ISO:0100.0505.0505.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
      *[IS-IS/18] 00:01:01
      Fictitious
PREFIX { Node { AS:64496 ISO:0100.0606.0606.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
      *[IS-IS/18] 00:01:01
      Fictitious
PREFIX { Node { AS:64496 ISO:0100.0707.0707.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
      *[IS-IS/18] 00:01:01
      Fictitious

```

```

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
  }/1152
          *[IS-IS/18] 00:01:01
          Fictitious

```

### show route te-ipv4-prefix-ip extensive

user@host>show route te-ipv4-prefix-ip 10.10.10.10 extensive

```

lsdist.0: 298 destinations, 298 routes (298 active, 0 holddown, 0 hidden)
  *IS-IS Preference: 15
    Level: 1
    Next hop type: Fictitious, Next hop index: 0
    Address: 0xala2ac4
    Next-hop reference count: 298
    Next hop:
    State:<Active NotInstall>
    Local AS: 64496
    Age: 7:58
    Validation State: unverified
    Task: IS-IS
    AS path: I
    Prefix SID: 1000, Flags: 0x40, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0101.0101.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
  }/1152 (1 entry, 0 announced)
  *IS-IS Preference: 18
    Level: 2
    Next hop type: Fictitious, Next hop index: 0
    Address: 0xala2ac4
    Next-hop reference count: 298
    Next hop:
    State: <Active NotInstall
    Local AS: 64496
    Age: 7:58
    Validation State: unverified
    Task: IS-IS
    AS path: I
    Prefix SID: 1000, Flags: 0xe0, Algo: 0>

PREFIX { Node { AS:64496 ISO:0100.0202.0202.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
  }/1152 (1 entry, 0 announced)
  *IS-IS Preference: 18
    Level: 2

```

```

        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 298
        Next hop:
        State: <Active NotInstall>
    Local AS:    64496
        Age: 7:58
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1000, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0303.0303.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
    }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 298
        Next hop:
        State: <Active NotInstall>
        Local AS:    64496
        Age: 7:58
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1000, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0404.0404.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
    }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 298
        Next hop:
        State: <Active NotInstall>
        Local AS:    64496
        Age: 7:58
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1000, Flags: 0xe0, Algo: 0

```



```

PREFIX { Node { AS:64496 ISO:0100.0505.0505.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
  }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
      Level: 2
      Next hop type: Fictitious, Next hop index: 0
      Address: 0xala2ac4
      Next-hop reference count: 298
      Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 7:58
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1000, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0606.0606.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
  }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
      Level: 2
      Next hop type: Fictitious, Next hop index: 0
      Address: 0xala2ac4
      Next-hop reference count: 298
      Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 7:58
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1000, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0707.0707.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
  }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
      Level: 2
      Next hop type: Fictitious, Next hop index: 0
      Address: 0xala2ac4
      Next-hop reference count: 298
      Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 7:58
        Validation State: unverified

```

```

Task: IS-IS
AS path: I
Prefix SID: 1000, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
  *IS-IS Preference: 18
    Level: 2
    Next hop type: Fictitious, Next hop index: 0
    Address: 0xala2ac4
    Next-hop reference count: 298
    Next hop:
    State: <Active NotInstall>
    Local AS: 64496
    Age: 7:58
    Validation State: unverified
    Task: IS-IS
    AS path: I
    Prefix SID: 1000, Flags: 0x40, Algo: 0

```

### show route te-ipv4-prefix-node-iso

user@host> show route te-ipv4-prefix-node-iso 0100.0a0a.0a0a.00

```

lsdist.0: 283 destinations, 283 routes (283 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L1:0
}/1152
    *[IS-IS/15] 00:05:20
    Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.1.1.1/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
    Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.2.2.2/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
    Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.3.3.3/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
    Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.4.4.4/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
    Fictitious

```

```

}/1152
    *[IS-IS/18] 00:05:20
        Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.5.5.5/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
        Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.6.6.6/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
        Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.7.7.7/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
        Fictitious
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152
    *[IS-IS/18] 00:05:20
        Fictitious

```

### show route te-ipv4-prefix-node-iso extensive

user@host> show route te-ipv4-prefix-node-iso 0100.0a0a.0a0a.00 extensive

```

lsdist.0: 283 destinations, 283 routes (283 active, 0 holddown, 0 hidden)
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L1:0
}/1152 (1 entry, 0 announced)
    *IS-IS Preference: 15
        Level: 1
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 283
        Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 6:47
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1000, Flags: 0x40, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.1.1.1/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)

```

```

*IS-IS Preference: 18
Level: 2
Next hop type: Fictitious, Next hop index: 0
Address: 0xala2ac4
Next-hop reference count: 283
Next hop:
State: <Active NotInstall>
Local AS: 64496
Age: 6:47
Validation State: unverified
Task: IS-IS
AS path: I
Prefix SID: 1001, Flags: 0xe0, Algo: 0

```

```

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.2.2.2/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)

```

```

*IS-IS Preference: 18
Level: 2
Next hop type: Fictitious, Next hop index: 0
Address: 0xala2ac4
Next-hop reference count: 283
Next hop:
State: <Active NotInstall>
Local AS: 64496
Age: 6:47
Validation State: unverified
Task: IS-IS
AS path: I
Prefix SID: 1002, Flags: 0xe0, Algo: 0

```

```

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.3.3.3/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)

```

```

*IS-IS Preference: 18
Level: 2
Next hop type: Fictitious, Next hop index: 0
Address: 0xala2ac4
Next-hop reference count: 283
Next hop:
State: <Active NotInstall>
Local AS: 64496
Age: 6:47
Validation State: unverified
Task: IS-IS
AS path: I

```

```

Prefix SID: 1003, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.4.4.4/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
  *IS-IS Preference: 18
    Level: 2
    Next hop type: Fictitious, Next hop index: 0
    Address: 0xala2ac4
    Next-hop reference count: 283
    Next hop:
    State: <Active NotInstall>
    Local AS: 64496
    Age: 6:47
    Validation State: unverified
    Task: IS-IS
    AS path: I
    Prefix SID: 1004, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.5.5.5/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
  *IS-IS Preference: 18
    Level: 2
    Next hop type: Fictitious, Next hop index: 0
    Address: 0xala2ac4
    Next-hop reference count: 283
    Next hop:
    State: <Active NotInstall>
    Local AS: 64496
    Age: 6:47
    Validation State: unverified
    Task: IS-IS
    AS path: I
    Prefix SID: 1005, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.6.6.6/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
  *IS-IS Preference: 18
    Level: 2
    Next hop type: Fictitious, Next hop index: 0
    Address: 0xala2ac4
    Next-hop reference count: 283
    Next hop:
    State: <Active NotInstall>
    Local AS: 64496

```

```

        Age: 6:47
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1006, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.7.7.7/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 283
        Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 6:47
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1007, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 283
        Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 6:47
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1000, Flags: 0x40, Algo: 0

```

**show route te-ipv4-prefix-node-iso detail**

user@host> **show route te-ipv4-prefix-node-iso 0100.0a0a.0a0a.00 detail**

```

lsdist.0: 283 destinations, 283 routes (283 active, 0 holddown, 0 hidden)
PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L1:0
  }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 15
      Level: 1
      Next hop type: Fictitious, Next hop index: 0
      Address: 0xala2ac4
      Next-hop reference count: 283
      Next hop:
      State: <Active NotInstall>
      Local AS: 64496
      Age: 6:54
      Validation State: unverified
      Task: IS-IS
      AS path: I
      Prefix SID: 1000, Flags: 0x40, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.1.1.1/32 } ISIS-L2:0
  }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
      Level: 2
      Next hop type: Fictitious, Next hop index: 0
      Address: 0xala2ac4
      Next-hop reference count: 283
      Next hop:
      State: <Active NotInstall>
      Local AS: 64496
      Age: 6:54
      Validation State: unverified
      Task: IS-IS
      AS path: I
      Prefix SID: 1001, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.2.2.2/32 } ISIS-L2:0
  }/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
      Level: 2
      Next hop type: Fictitious, Next hop index: 0
      Address: 0xala2ac4
      Next-hop reference count: 283
      Next hop:
      State: <Active NotInstall>
      Local AS: 64496
      Age: 6:54

```

```

        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1002, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.3.3.3/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 283
        Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 6:54
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1003, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.4.4.4/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 283
        Next hop:
        State: <Active NotInstall>
        Local AS: 64496
        Age: 6:54
        Validation State: unverified
        Task: IS-IS
        AS path: I
        Prefix SID: 1004, Flags: 0xe0, Algo: 0

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.5.5.5/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)
    *IS-IS Preference: 18
        Level: 2
        Next hop type: Fictitious, Next hop index: 0
        Address: 0xala2ac4
        Next-hop reference count: 283

```



```

Next hop:
State: <Active NotInstall>
Local AS: 64496
Age: 6:54
Validation State: unverified
Task: IS-IS
AS path: I
Prefix SID: 1005, Flags: 0xe0, Algo: 0

```

```

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.6.6.6/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)

```

```

*IS-IS Preference: 18
Level: 2
Next hop type: Fictitious, Next hop index: 0
Address: 0xala2ac4
Next-hop reference count: 283
Next hop:
State: <Active NotInstall>
Local AS: 64496
Age: 6:54
Validation State: unverified
Task: IS-IS
AS path: I
Prefix SID: 1006, Flags: 0xe0, Algo: 0

```

```

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.7.7.7/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)

```

```

*IS-IS Preference: 18
Level: 2
Next hop type: Fictitious, Next hop index: 0
Address: 0xala2ac4
Next-hop reference count: 283
Next hop:
State: <Active NotInstall>
Local AS: 64496
Age: 6:54
Validation State: unverified
Task: IS-IS
AS path: I
Prefix SID: 1007, Flags: 0xe0, Algo: 0

```

```

PREFIX { Node { AS:64496 ISO:0100.0a0a.0a0a.00 } { IPv4:10.10.10.10/32 } ISIS-L2:0
}/1152 (1 entry, 0 announced)

```

```

*IS-IS Preference: 18

```

```

Level: 2
Next hop type: Fictitious, Next hop index: 0
Address: 0xala2ac4
Next-hop reference count: 283
Next hop:
State: <Active NotInstall>
Local AS: 64496
Age: 6:54
Validation State: unverified
Task: IS-IS
AS path: I
Prefix SID: 1000, Flags: 0x40, Algo: 0

```

### show route rib-sharding junos-bgpshard14

user@host> show route rib-sharding junos-bgpshard14

```

inet.0: 54 destinations, 54 routes (54 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2.2.50.8/30      *[Direct/0] 00:29:15
                  > via eth9.101
2.2.50.9/32      *[Local/0] 00:29:15
                  Local via eth9.101
2.2.51.8/30      *[Direct/0] 00:29:15
                  > via eth9.102@
2.2.51.9/32      *[Local/0] 00:29:15
                  Local via eth9.102@
2.2.52.8/30      *[Direct/0] 00:29:15
                  > via eth8.103
2.2.52.9/32      *[Local/0] 00:29:15
                  Local via eth8.103
2.2.54.8/30      *[Direct/0] 00:29:15
                  > via eth8.104@
2.2.54.9/32      *[Local/0] 00:29:15
                  Local via eth8.104@
2.2.55.8/30      *[Direct/0] 00:29:15
                  > via eth6.105
2.2.55.9/32      *[Local/0] 00:29:15
                  Local via eth6.105
2.2.56.8/30      *[Direct/0] 00:29:15
                  > via eth6.106@
2.2.56.9/32      *[Local/0] 00:29:15
                  Local via eth6.106@

```

```

2.2.57.8/30      *[Direct/0] 00:29:15
                  >  via eth7.107
2.2.57.9/32      *[Local/0] 00:29:15
                  Local via eth7.107
2.2.58.8/30      *[Direct/0] 00:29:15
                  >  via eth7.108@
2.2.58.9/32      *[Local/0] 00:29:15
                  Local via eth7.108@
3.2.56.8/30      *[Direct/0] 00:23:38
                  >  via eth11.115
3.2.56.9/32      *[Local/0] 00:23:38
                  Local via eth11.115
3.2.57.8/30      *[Direct/0] 00:23:38
                  >  via eth11.116
3.2.57.9/32      *[Local/0] 00:23:38
                  Local via eth11.116
3.12.50.8/30     *[Direct/0] 00:23:38
                  >  via eth13.109
3.12.50.9/32     *[Local/0] 00:23:38
                  Local via eth13.109
3.12.51.8/30     *[Direct/0] 00:23:38
                  >  via eth13.110
3.12.51.9/32     *[Local/0] 00:23:38
                  Local via eth13.110
3.12.52.8/30     *[Direct/0] 00:23:38
                  >  via eth10.111
3.12.52.9/32     *[Local/0] 00:23:38
                  Local via eth10.111
3.12.53.8/30     *[Direct/0] 00:23:38
                  >  via eth10.112
3.12.53.9/32     *[Local/0] 00:23:38
                  Local via eth10.112
3.12.55.8/30     *[Direct/0] 00:23:38
                  >  via eth12.113
3.12.55.9/32     *[Local/0] 00:23:38
                  Local via eth12.113
3.12.56.8/30     *[Direct/0] 00:23:38
                  >  via eth12.114
3.12.56.9/32     *[Local/0] 00:23:38
                  Local via eth12.114
10.0.0.0/24      *[Direct/0] 1d 01:57:27
                  >  via eth5.100
10.0.0.1/32      *[Local/0] 1d 01:57:27
                  Local via eth5.100

```

```

10.216.160.0/21    *[Direct/0] 1d 01:57:27
                  > via eth0,V
10.216.161.169/32 *[Local/0] 1d 01:57:27
                  Local via eth0,V
12.2.45.0/24      *[Direct/0] 16:49:27
                  > via eth4,V
12.2.45.9/32      *[Local/0] 16:49:27
                  Local via eth4,V
12.2.46.0/24      *[Direct/0] 1d 01:57:27
                  > via eth2,V
12.2.46.9/32      *[Local/0] 1d 01:57:27
                  Local via eth2,V
20.0.0.0/24       *[Direct/0] 1d 01:57:27
                  > via eth5.200
20.0.0.1/32       *[Local/0] 1d 01:57:27
                  Local via eth5.200
20.255.255.10/32  *[Direct/0] 1d 01:57:27
                  > via lo.0
20.255.255.11/32  *[Direct/0] 1d 01:57:27
                  > via lo.0
20.255.255.12/32  *[Direct/0] 1d 01:57:27
                  > via lo.0
20.255.255.13/32  *[Direct/0] 1d 01:57:27
                  > via lo.0
20.255.255.14/32  *[Direct/0] 1d 01:57:27
                  > via lo.0
20.255.255.15/32  *[Direct/0] 1d 01:57:27
                  > via lo.0
20.255.255.16/32  *[Direct/0] 1d 01:57:27
                  > via lo.0
30.0.0.0/24       *[Direct/0] 1d 01:57:27
                  > via eth3,V
30.0.0.1/32       *[Local/0] 1d 01:57:27
                  Local via eth3,V
172.17.0.1/32     *[Local/0] 1d 01:57:27
                  Reject
172.18.0.1/32     *[Local/0] 1d 01:57:27
                  Reject
192.168.2.254/32  *[Local/0] 1d 01:57:27
                  Reject

iso.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

```

47.0005.1921.6810.0012/72
    *[Direct/0] 1d 01:57:27
    > via lo.0

inet6.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

fe80::42:ffff:fe7a:26ec/128
    *[Local/0] 1d 01:57:27
    Reject
fe80::ec4:7aff:feda:43c8/128
    *[Local/0] 1d 01:57:27
    Local via eth0,V
fe80::92e2:baff:fecc:3ed4/128
    *[Local/0] 00:23:38
    Local via eth12.114
fe80::92e2:baff:fecc:3ed5/128
    *[Local/0] 00:23:38
    Local via eth13.110
fe80::92e2:baff:fecc:a024/128
    *[Local/0] 16:49:27
    Local via eth4,V
fe80::92e2:baff:fecc:a025/128
    *[Local/0] 1d 01:57:27
    Local
fe80::92e2:baff:fecc:a030/128
    *[Local/0] 00:29:15
    Local via eth6.106@
fe80::92e2:baff:fecc:a031/128
    *[Local/0] 00:29:15
    Local via eth7.108@
fe80::92e2:baff:fecc:a038/128
    *[Local/0] 1d 01:57:27
    Local via eth2,V
fe80::92e2:baff:fecc:a039/128
    *[Local/0] 1d 01:57:27
    Local via eth3,V
fe80::92e2:baff:fecc:a050/128
    *[Local/0] 00:29:15
    Local via eth8.104@
fe80::92e2:baff:fecc:a051/128
    *[Local/0] 00:29:15
    Local via eth9.102@
fe80::92e2:baff:fecc:a054/128

```

```
*[Local/0] 00:23:38
    Local via eth10.112
fe80::92e2:baff:fedc:a055/128
*[Local/0] 00:23:38
    Local via eth11.116
```