



Ethernet Switching Feature Guide for the QFX Series

Release

14.1X53



Modified: 2016-11-03

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

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

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

Ethernet Switching Feature Guide for the QFX Series

14.1X53

Copyright © 2016, Juniper Networks, Inc.

All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

	About the Documentation	xvii
	Documentation and Release Notes	xvii
	Supported Platforms	xvii
	Using the Examples in This Manual	xvii
	Merging a Full Example	xviii
	Merging a Snippet	xviii
	Documentation Conventions	xix
	Documentation Feedback	xxi
	Requesting Technical Support	xxi
	Self-Help Online Tools and Resources	xxi
	Opening a Case with JTAC	xxii
Part 1	Ethernet Ring Protection	
Chapter 1	Understanding Ethernet Ring Protection	3
	Ethernet Ring Protection Switching Overview	3
	Understanding Ethernet Ring Protection Switching Functionality	4
	Acronyms	4
	Ring Nodes	5
	Ring Node States	5
	Default Logging of Basic State Transitions on EX Series Switches	5
	Failure Detection	6
	Logical Ring	6
	FDB Flush	6
	Traffic Blocking and Forwarding	6
	RAPS Message Blocking and Forwarding	6
	Dedicated Signaling Control Channel	8
	RAPS Message Termination	8
	Multiple Rings	8
	Node ID	8
	Bridge Domains with the Ring Port (MX Series Routers Only)	8
	Configuring Ethernet Ring Protection Switching (CLI Procedure)	9
	Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS	12
Part 2	Bridging and VLANs	
Chapter 2	Using Bridging and VLANs	31
	Overview of Layer 2 Networking	31
	Understanding Layer 2 Broadcasting	34
	Layer 2 Learning and Forwarding for VLANs Overview	34

Understanding Bridging and VLANs	35
History of VLANs	35
How Bridging of VLAN Traffic Works	36
Packets Are Either Tagged or Untagged	37
Switch Interface Modes—Access, Trunk, or Tagged Access	37
Access Mode	38
Trunk Mode	38
Trunk Mode and Native VLAN	38
Tagged-Access Mode	39
Additional Advantages of Using VLANs	39
Maximum VLANs and VLAN Members Per Switch	40
A Default VLAN Is Configured on Most Switches	41
Assigning Traffic to VLANs	41
Assign VLAN Traffic According to the Interface Port Source	41
Assign VLAN Traffic According to the Source MAC Address	42
Forwarding VLAN Traffic	42
VLANs Communicate with Integrated Routing and Bridging Interfaces or Routed VLAN Interfaces	42
Example: Setting Up Basic Bridging and a VLAN on the QFX Series	43
Example: Setting Up Bridging with Multiple VLANs	60
Configuring VLANs	66
Configuring the Native VLAN Identifier (CLI Procedure)	67
Creating a Series of Tagged VLANs	69
Understanding Integrated Routing and Bridging	70
Configuring IRB Interfaces	72
Example: Configuring Routing Between VLANs on One Switch	73
Excluding a Routed VLAN Interface from State Calculations	79
Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure)	81
Configuring Static ARP Entries	81
Understanding Multiple VLAN Registration Protocol (MVRP)	82
QFabric Requirements	82
MVRP Operations	83
MRP Timers Control MVRP Updates	83
MVRP Uses MRP Messages to Transmit Switch and VLAN States	84
Troubleshooting Ethernet Switching	84

Part 3

Chapter 3

MAC Addresses

Using MAC Addresses	89
Introduction to the Media Access Control (MAC) Layer 2 Sublayer	89
Understanding MAC Learning	90
Disabling MAC Learning	90
Example: Disabling MAC Learning	91
Configuring MAC Notification (CLI Procedure)	92
Enabling MAC Notification	93
Disabling MAC Notification	93

	Setting the MAC Notification Interval	93
	Verifying That MAC Notification Is Working Properly	93
	Configuring MAC Limiting (CLI Procedure)	94
	Limiting the Number of MAC Addresses Learned by an Interface	95
	Limiting the Number of MAC Addresses Learned by a VLAN	95
	Configuring MAC Table Aging	96
Part 4	Spanning Trees	
Chapter 4	Using Spanning Trees	99
	Overview of Spanning-Tree Protocols	100
	Understanding Spanning Tree Protocols on a QFabric System	100
	Understanding MSTP	101
	Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches	101
	Understanding RSTP	121
	Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches	123
	Configuring RSTP (CLI Procedure)	139
	Understanding VSTP	140
	Example: Configuring VSTP on QFX Series Switches and EX4600 Switches . . .	141
	Understanding BPDU Protection for STP, RSTP, and MSTP	146
	Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations on EX Series Switches	147
	Understanding Loop Protection for STP, RSTP, VSTP, and MSTP	151
	Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree	152
	Understanding Root Protection for STP, RSTP, VSTP, and MSTP	156
	Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees	157
Part 5	Q-in-Q Tunneling	
Chapter 5	Using Q-in-Q Tunneling	165
	Understanding Q-in-Q Tunneling	165
	How Q-in-Q Tunneling Works	166
	How VLAN Translation Works	166
	Using Dual VLAN Tag Translation	167
	Sending and Receiving Untagged Packets	167
	Disabling MAC Address Learning	168
	Mapping C-VLANs to S-VLANs	168
	All-in-One Bundling	168
	Many-to-Many Bundling	168
	Mapping a Specific Interface	169
	Constraints for Q-in-Q Tunneling and VLAN Translation	169
	Configuring Q-in-Q Tunneling	170
	Using the Different Mapping Methods	171
	Configuring All-in-One Bundling	172
	Configuring Many-to-Many Bundling	173

	Configuring a Specific Interface Mapping with VLAN ID Translation Option	176
	Configuring All-in-One Bundling	178
	Configuring Many-to-Many Bundling	180
	Configuring a Specific Interface Mapping with VLAN ID Translation Option	183
	Setting Up a Dual VLAN Tag Translation Configuration	185
Part 6	Private VLANs	
Chapter 6	Using Private VLANs	189
	Understanding Private VLANs	189
	Typical Structure and Primary Application of PVLANS	190
	Using 802.1Q Tags to Identify Packets	191
	Efficient Use of IP Addresses	192
	PVLAN Port Types	192
	Limitations of Private VLANs	194
	Understanding PVLAN Traffic Flows Across Multiple Switches	194
	Community VLAN Sending Untagged Traffic	194
	Isolated VLAN Sending Untagged Traffic	195
	PVLAN Tagged Traffic Sent on a Promiscuous Port	196
	Using an IRB Interface in a Private VLAN	198
	Configuring an IRB Interface in a Private VLAN	198
	IRB Interface Limitation in a PVLAN	198
	Example: Configuring a Private VLAN Spanning Multiple Switches With an IRB Interface	199
	Understanding Egress Firewall Filters with PVLANS	212
	Creating a Private VLAN on a Single Switch (CLI Procedure)	214
	Creating a Private VLAN Spanning Multiple Switches (CLI Procedure)	216
	Example: Configuring a Private VLAN on a Single Switch	217
	Verifying That a Private VLAN Is Working	221
Part 7	Proxy ARP	
Chapter 7	Using Proxy ARP	229
	Understanding Proxy ARP	229
	What Is ARP?	229
	Proxy ARP Overview	229
	Best Practices for Proxy ARP	230
	Configuring Proxy ARP (CLI Procedure)	231
	Verifying That Proxy ARP Is Working Correctly	231
Part 8	Reflective Relay	
Chapter 8	Using Reflective Relay	235
	Understanding Reflective Relay for Use with VEPA Technology	235
	VEPA	235
	Reflective Relay	235
	Configuring Reflective Relay	236
	Example: Configuring Reflective Relay for Use with VEPA Technology	237

Part 9	Unified Forwarding Table	
Chapter 9	Using the Unified Forwarding Table	245
	Understanding the Unified Forwarding Table	245
	Using the Unified Forwarding Table to Optimize Address Storage	245
	MAC Address and Host Address Memory Allocation	245
	Using the LPM Table with Junos OS 13.2X51-D10	247
	LPM Table Memory Allocation	247
	Configuring the Unified Forwarding Table	248
	Configuring an Address-Storage Profile	248
	Configuring the LPM Allocation	249
	Configuring the LPM Table With Junos OS 13.2X51-D10 and 13.2X52-D10	250
	Configuring the LPM Table With Junos OS 13.2x51-D15 and Later	251
Part 10	Configuration Statements and Operational Commands	
Chapter 10	Ethernet Ring Protection Configuration Statements	257
	control-channel	258
	control-vlan	259
	data-channel	260
	east-interface	261
	ethernet-ring	262
	guard-interval	263
	hold-interval (Protection Group)	264
	protection-group	265
	restore-interval	266
	ring-protection-link-end	267
	ring-protection-link-owner	267
	west-interface	268
Chapter 11	VLAN Configuration Statements	269
	[edit vlans] Configuration Statement Hierarchy on the QFX Series	270
	Supported Statements in the [edit vlans] Hierarchy Level	270
	Unsupported Statements in the [edit vlans] Hierarchy Level	272
	autostate-exclude	273
	description (VLAN)	274
	dhcp-relay	275
	filter (VLANs)	280
	forwarding-options	281
	interface (VLANs)	286
	interface-mac-limit	287
	interface-mode	289
	irb (Interfaces)	291
	isolated-vlan	294
	l3-interface (VLAN)	295
	mac (Static MAC-Based VLANs)	296
	members	297
	native-vlan-id	298
	packet-action	299

	port-mode	302
	private-vlan	303
	service-id	304
	switch-options	305
	static (Static MAC-Based VLANs)	306
	static-mac	306
	vlan-id (VLANs)	307
	vlan-id-list	308
	vlan-rewrite	309
	vlan-tagging	310
	vlan-tags	311
	vlangs	312
Chapter 12	MAC Address Configuration Statements	315
	mac-limit	315
	mac-notification	316
	mac-statistics	317
	mac-table-aging-time	318
	mac-table-size	319
	notification-interval	320
Chapter 13	Private VLAN Configuration Statements	321
	extend-secondary-vlan-id	321
	isolated	322
	isolation-vlan-id	322
	primary-vlan	323
	promiscuous	323
	vlangs	324
Chapter 14	STP Configuration Statements	327
	bpdu-block	328
	bpdu-block-on-edge	329
	bpdu-timeout-action	330
	configuration-name	331
	cost	332
	bpdu-block	333
	bpdu-block-on-edge	333
	bpdu-timeout-action	334
	configuration-name	335
	cost	336
	disable-timeout (Spanning Trees)	337
	edge	338
	force-version (IEEE 802.1D STP)	339
	forward-delay	340
	hello-time	341
	interface (BPDU)	342
	max-age	343
	max-hops	344
	mode (Protocols STP)	345
	mstp	346

	no-root-port	347
	priority (Protocols STP)	348
	protocol	349
	protocols (STP Type)	350
	revision-level	351
	rstp	352
	traceoptions (Spanning Tree)	353
	vlan (VSTP)	356
	vlan-group	357
	vstp	358
Chapter 15	Q-in-Q Configuration Statements	361
	flexible-vlan-tagging	362
	input-vlan-map	363
	native-vlan-id	364
	output-vlan-map	365
	pop	366
	push	367
	swap	368
	vlan-id-list	369
Chapter 16	Reflective Relay Configuration Statement	371
	reflective-relay	371
Chapter 17	Unified Forwarding Table Configuration Statements	373
	forwarding-options (chassis)	374
	num-65-127-prefix	376
	prefix-65-127-disable	376
Chapter 18	Bridging and VLANs Monitoring Commands	377
	clear ethernet-switching table	378
	show ethernet-switching interface	380
	show ethernet-switching interfaces	383
	show ethernet-switching table	387
	show system statistics arp	395
	show vlans	396
Chapter 19	MAC Address Operational Commands	405
	show ethernet-switching mac-learning-log	406
	show ethernet-switching mac-notification	408
	show ethernet-switching statistics aging	410
	show ethernet-switching statistics mac-learning	412
Chapter 20	Spanning Tree Monitoring Commands	417
	clear spanning-tree statistics	418
	show spanning-tree bridge	419
	show spanning-tree interface	424
	show spanning-tree mstp configuration	430
	show spanning-tree statistics	432

List of Figures

Part 1	Ethernet Ring Protection	
Chapter 1	Understanding Ethernet Ring Protection	3
	Figure 1: Protocol Packets from the Network to the Router	6
	Figure 2: Protocol Packets from the Router or Switch to the Network	7
	Figure 3: Ethernet Ring Protection Switching Example	13
Part 2	Bridging and VLANs	
Chapter 2	Using Bridging and VLANs	31
	Figure 4: IRB with One Switch	74
Part 4	Spanning Trees	
Chapter 4	Using Spanning Trees	99
	Figure 5: Network Topology for MSTP	103
	Figure 6: Network Topology for RSTP	124
	Figure 7: BPDU Protection Topology	148
	Figure 8: Network Topology for Loop Protection	154
	Figure 9: Network Topology for Root Protection	159
Part 6	Private VLANs	
Chapter 6	Using Private VLANs	189
	Figure 10: Subdomains in a PVLAN	190
	Figure 11: PVLAN Spanning Multiple Switches	191
	Figure 12: Community VLAN Sends Untagged Traffic	195
	Figure 13: Isolated VLAN Sends Untagged Traffic	196
	Figure 14: PVLAN Tagged Traffic Sent on a Promiscuous Port	197
	Figure 15: PVLAN Topology Spanning Multiple Switches with an IRB Interface	200
	Figure 16: Topology of a Private VLAN on a Single EX Series Switch	219
Part 8	Reflective Relay	
Chapter 8	Using Reflective Relay	235
	Figure 17: Reflective Relay Topology	239

List of Tables

	About the Documentation	xvii
	Table 1: Notice Icons	xix
	Table 2: Text and Syntax Conventions	xix
Part 1	Ethernet Ring Protection	
Chapter 1	Understanding Ethernet Ring Protection	3
	Table 3: Components to Configure for This Example	14
Part 2	Bridging and VLANs	
Chapter 2	Using Bridging and VLANs	31
	Table 4: Components of the Basic Bridging Configuration Topology	44
	Table 5: Components of the Multiple VLAN Topology	61
	Table 6: Sample IRB Values	71
	Table 7: Number of Supported IRBs/RVIs by Platform	71
	Table 8: Components of the Multiple VLAN Topology	74
	Table 9: MVRP VLAN Requirements for Node Devices	83
Part 4	Spanning Trees	
Chapter 4	Using Spanning Trees	99
	Table 10: Components of the Topology for Configuring MSTP on EX Series Switches	103
	Table 11: Components of the Topology for Configuring RSTP	125
	Table 12: Interfaces of the Topology for Configuring VSTP	142
	Table 13: Components of the Topology for Configuring BPDU Protection on EX Series Switches	148
	Table 14: Topology for Configuring Loop Protection on the QFX Series	154
	Table 15: Topology for Configuring Root Protection on the QFX Series	159
Part 5	Q-in-Q Tunneling	
Chapter 5	Using Q-in-Q Tunneling	165
	Table 16: Operations Added with Dual VLAN Tag Rewrite	167
Part 6	Private VLANs	
Chapter 6	Using Private VLANs	189
	Table 17: PVLAN Requirements for 802.1Q Tags	192
	Table 18: PVLAN Ports and Layer 2 Connectivity	193

	Table 19: Components of Switch 1 in the Topology for Configuring a PVLAN Spanning Multiple Devices	201
	Table 20: Components of Switch 2 in the Topology for Configuring a PVLAN Spanning Multiple Devices	201
	Table 21: Components of Switch 3 in the Topology for Configuring a PVLAN Spanning Multiple Devices	202
	Table 22: Interfaces of the Topology for Configuring a PVLAN	218
	Table 23: VLAN IDs in the Topology for Configuring a PVLAN	218
Part 8	Reflective Relay	
Chapter 8	Using Reflective Relay	235
	Table 24: Components of the Topology for Configuring Reflective Relay	239
Part 9	Unified Forwarding Table	
Chapter 9	Using the Unified Forwarding Table	245
	Table 25: Unified Forwarding Table Profiles	246
	Table 26: Example Host Table Combinations Using l2-profile-one	246
	Table 27: Example LPM Table Combinations Using l2-profile-one Using Junos OS 13.2X51-D10	247
	Table 28: Unified Forwarding Table Profiles	248
	Table 29: Example LPM Table Combinations Using l2-and l3 Profiles With Junos OS 13.2X51-D10 and 13.2X52-D10	250
	Table 30: LPM Table Combinations for l2 and l3 profiles With Junos OS 13.2X51-D15	252
	Table 31: lpm-profile with unicast-in-lpm Option	253
Part 10	Configuration Statements and Operational Commands	
Chapter 11	VLAN Configuration Statements	269
	Table 32: Unsupported [edit vlans] Configuration Statements on EX Series Switches	272
Chapter 17	Unified Forwarding Table Configuration Statements	373
	Table 33: Unified Forwarding Table Profiles	375
Chapter 18	Bridging and VLANs Monitoring Commands	377
	Table 34: show ethernet-switching interface Output Fields	380
	Table 35: show ethernet-switching interfaces Output Fields	383
	Table 36: show ethernet-switching table Output Fields	387
	Table 37: show vlans Output Fields	397
Chapter 19	MAC Address Operational Commands	405
	Table 38: show ethernet-switching mac-learning-log Output Fields	406
	Table 39: show ethernet-switching mac-notification Output Fields	408
	Table 40: show ethernet-switching statistics aging Output Fields	410
	Table 41: show ethernet-switching statistics mac-learning Output Fields	413
Chapter 20	Spanning Tree Monitoring Commands	417
	Table 42: show spanning-tree bridge Output Fields	419

Table 43: show spanning-tree Interface Output Fields	424
Table 44: show spanning-tree mstp configuration Output Fields	430
Table 45: show spanning-tree statistics Output Fields	432

About the Documentation

- [Documentation and Release Notes on page xvii](#)
- [Supported Platforms on page xvii](#)
- [Using the Examples in This Manual on page xvii](#)
- [Documentation Conventions on page xix](#)
- [Documentation Feedback on page xxi](#)
- [Requesting Technical Support on page xxi](#)

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 <http://www.juniper.net/techpubs/>.

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

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

Supported Platforms

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

- [QFX Series standalone switches](#)

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

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

Table 1: Notice Icons

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

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

Table 2: Text and Syntax Conventions

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

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

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

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

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

Documentation Feedback

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

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

Requesting Technical Support

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

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

Self-Help Online Tools and Resources

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

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

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

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

Opening a Case with JTAC

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

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

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

PART 1

Ethernet Ring Protection

- [Understanding Ethernet Ring Protection on page 3](#)

CHAPTER 1

Understanding Ethernet Ring Protection

- [Ethernet Ring Protection Switching Overview on page 3](#)
- [Understanding Ethernet Ring Protection Switching Functionality on page 4](#)
- [Configuring Ethernet Ring Protection Switching \(CLI Procedure\) on page 9](#)
- [Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12](#)

Ethernet Ring Protection Switching Overview

Ethernet ring protection switching (ERPS) helps achieve high reliability and network stability. Links in the ring will never form loops that fatally affect the network operation and services availability. The basic idea of an Ethernet ring is to use one specific link to protect the whole ring. This special link is called a *ring protection link (RPL)*. If no failure happens in other links of the ring, the RPL blocks the traffic and is not used. The RPL is controlled by a special node called an *RPL owner*. There is only one RPL owner in a ring. The RPL owner is responsible for blocking traffic over the RPL. Under ring failure conditions, the RPL owner is responsible for unblocking traffic over the RPL. A ring failure results in protection switching of the RPL traffic. An automatic protection switching (APS) protocol is used to coordinate the protection actions over the ring. Protection switching blocks traffic on the failed link and unblocks the traffic on the RPL. When the failure clears, revertive protection switching blocks traffic over the RPL and unblocks traffic on the link on which the failure is cleared.

The following standards provide detailed information on Ethernet ring protection switching:

- IEEE 802.1Q - 1998
- IEEE 802.1D - 2004
- IEEE 802.1Q - 2003
- Draft ITU-T Recommendation G.8032/Y.1344, *Ethernet Ring protection switching*
- ITU-T Y.1731, *OAM functions and mechanisms for Ethernet-based networks*

For additional information on configuring Ethernet ring protection switching on EX Series switches, see *Example: Configuring Ethernet Ring Protection Switching on EX Series Switches*.

For additional information on configuring Ethernet ring protection switching on MX Series routers, see the *Layer 2 Configuration Guide* for a complete example of Ethernet rings and information about STP loop avoidance and prevention.

**Related
Documentation**

- [Understanding Ethernet Ring Protection Switching Functionality on page 4](#)
- [Configuring Ethernet Ring Protection Switching](#)
- [Example: Ethernet Ring Protection Switching Configuration on MX Routers](#)
- [Example: Configuring Ethernet Ring Protection Switching on EX Series Switches](#)
- [Ethernet Interfaces Feature Guide for Routing Devices](#)

Understanding Ethernet Ring Protection Switching Functionality

- [Acronyms on page 4](#)
- [Ring Nodes on page 5](#)
- [Ring Node States on page 5](#)
- [Default Logging of Basic State Transitions on EX Series Switches on page 5](#)
- [Failure Detection on page 6](#)
- [Logical Ring on page 6](#)
- [FDB Flush on page 6](#)
- [Traffic Blocking and Forwarding on page 6](#)
- [RAPS Message Blocking and Forwarding on page 6](#)
- [Dedicated Signaling Control Channel on page 8](#)
- [RAPS Message Termination on page 8](#)
- [Multiple Rings on page 8](#)
- [Node ID on page 8](#)
- [Bridge Domains with the Ring Port \(MX Series Routers Only\) on page 8](#)

Acronyms

The following acronyms are used in the discussion about Ethernet ring protection switching (ERPS):

- MA—Maintenance association
- MEP—Maintenance association end point
- OAM—Operations, administration, and management (Ethernet ring protection switching uses connectivity fault management daemon)
- FDB—MAC forwarding database
- STP—Spanning Tree Protocol
- RAPS—Ring automatic protection switching

- WTR—Wait to restore
- RPL—Ring protection link

Ring Nodes

Multiple nodes are used to form a ring. There are two different node types:

- Normal node—The node has no special role on the ring.
- RPL owner node—The node owns the RPL and blocks or unblocks traffic over the RPL. This node also initiates the RAPS message.

Ring Node States

There are three different states for each node of a specific ring:

- init—Not a participant of a specific ring.
- idle—No failure on the ring; the node is performing normally. For a normal node, traffic is unblocked on both ring ports. For the RPL owner, traffic is blocked on the ring port that connects to the RPL and unblocked on the other ring port.
- protection—A failure occurred on the ring. For a normal node, traffic is blocked on the ring port that connects to the failing link and unblocked on working ring ports. For the RPL owner, traffic is unblocked on both ring ports if they connect to non-failure links.

There can be only one RPL owner for each ring. The user configuration must guarantee this, because the APS protocol cannot check this.

Default Logging of Basic State Transitions on EX Series Switches



NOTE: This section applies only to EX Series switches that run Junos OS that does not support the Enhanced Layer 2 Software (ELS) configuration style.

Starting with Junos OS Release 14.1X53-D15, EX Series switches automatically log basic state transitions for the ERPS protocol. No configuration is required to initiate this logging. Basic state transitions include ERPS interface transitions from up to down, and down to up; and ERPS state transitions from idle to protection, and protection to idle.

The basic state transitions are logged in a single file named **erp-default**, which resides in the **/var/log** directory of the switch. The maximum size of this file is 15 MB.

Default logging for ERPS can capture initial ERPS interface and state transitions, which can help you troubleshoot issues that occur early in the ERPS protocol startup process. However, if more robust logging is needed, you can enable traceoptions for ERPS by entering the **traceoptions** statement in the **[edit protocols protection-group]** hierarchy.

Be aware that for ERPS, only default logging or traceoptions can be active at a time on the switch. That is, default logging for ERPS is automatically enabled and if you enable traceoptions for ERPS, the switch automatically disables default logging. Conversely, if you disable traceoptions for ERPS, the switch automatically enables default logging.

Failure Detection

Ethernet ring operation depends on quick and accurate failure detection. The failure condition *signal failure (SF)* is supported. For SF detection, an Ethernet continuity check MEP must be configured for each ring link. For fast protection switching, a 10-ms transmission period for this MEP group is supported. OAM monitors the MEP group's MA and reports SF or SF clear events to the Ethernet ring control module. For this MEP group, the action profile must be configured to update the interface device IFF_LINKDOWN flag. OAM updates the IFF_LINKDOWN flag to notify the Ethernet ring control module.

Logical Ring

This feature currently supports only the physical ring, which means that two adjacent nodes of a ring must be physically connected and the ring must operate on the physical interface, not the VLAN.

FDB Flush

When ring protection switching occurs, normally an *FDB flush* is executed. The Ethernet ring control module uses the same mechanism as the STP to trigger the FDB flush. The Ethernet ring control module controls the ring port physical interface's default STP index to execute the FDB flush.

Traffic Blocking and Forwarding

Ethernet ring control uses the same mechanism as the STP to control forwarding or discarding of user traffic. The Ethernet ring control module sets the ring port physical interface default STP index state to forwarding or discarding in order to control user traffic.

RAPS Message Blocking and Forwarding

The router or switch treats the ring automatic protection switching (RAPS) message the same as it treats user traffic for forwarding RAPS messages between two ring ports. The ring port physical interface default STP index state also controls forwarding RAPS messages between the two ring ports. Other than forwarding RAPS messages between the two ring ports, as shown in [Figure 1 on page 6](#), the system also needs to forward the RAPS message between the CPU (Ethernet ring control module) and the ring port. This type of forwarding does not depend on the ring port physical interfaces' STP index state. The RAPS message is always sent by the router or switch through the ring ports, as shown in [Figure 2 on page 7](#). A RAPS message received from a discarding ring port is sent to the Ethernet ring control module, but is not sent to the other ring port.

Figure 1: Protocol Packets from the Network to the Router

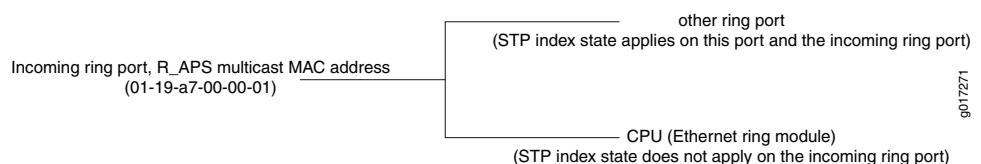
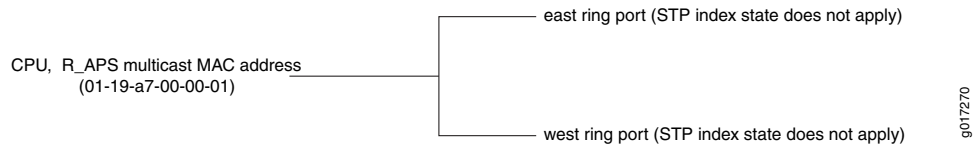


Figure 2: Protocol Packets from the Router or Switch to the Network



Juniper Networks switches and Juniper Networks routers use different methods to achieve these routes.

The switches use forwarding database entries to direct the RAPS messages. The forwarding database entry (keyed by the RAPS multicast address and VLAN) has a composite next hop associated with it—the composite next hop associates the two ring interfaces with the forwarding database entry and uses the split horizon feature to prevent sending the packet out on the interface that it is received on. This is an example of the forwarding database entry relating to the RAPS multicast MAC (a result of the **show ethernet-switching table detail** command):

```

VLAN: v1, Tag: 101, MAC: 01:19:a7:00:00:01, Interface: ERP
Interfaces:                               ge-0/0/9.0, ge-0/0/3.0
Type: Static
Action: Mirror
Nexthop index: 1333
  
```

The routers use an implicit filter to achieve ERP routes. Each implicit filter binds to a bridge domain. Therefore, the east ring port control channel and the west ring port control channel of a particular ring instance must be configured to the same bridge domain. For each ring port control channel, a filter term is generated to control RAPS message forwarding. The filter number is the same as the number of bridge domains that contain the ring control channels. If a bridge domain contains control channels from multiple rings, the filter related to this bridge domain will have multiple terms and each term will relate to a control channel. The filter has command parts and control-channel related parts, as follows:

- Common terms:
 - term 1: if [Ethernet type is not OAM Ethernet type (0x8902)]
 { accept packet }
 - term 2: if [source MAC address belongs to this bridge]
 { drop packet, our packet loop through the ring and come back to home }
 - term 3: if [destination is the RAPS PDU multicast address(0x01,0x19,0xa7,0x00,0x00,0x01) AND[ring port STP status is DISCARDING]
 { send to CPU }
- Control channel related terms:
 - if [destination is the RAPS PDU multicast address(0x01,0x19,0xa7,0x00,0x00,0x01) AND[ring port STP status is FORWARDING] AND [Incoming interface IFL equal to control channel IFL]
 { send packet to CPU and send to the other ring port }
 default term: accept packet.

Dedicated Signaling Control Channel

For each ring port, a dedicated signaling control channel with a dedicated VLAN ID must be configured. In Ethernet ring configuration, only this control logical interface is configured and the underlying physical interface is the physical ring port. Each ring requires that two control physical interfaces be configured. These two logical interfaces must be configured in a bridge domain for routers (or the same VLAN for switches) in order to forward RAPS protocol data units (PDUs) between the two ring control physical interfaces. If the router control channel logical interface is not a trunk port, only control logical interfaces will be configured in ring port configuration. If this router control channel logical interface is a trunk port, in addition to the control channel logical interfaces, a dedicated VLAN ID must be configured for routers. For switches, always specify either a VLAN name or VLAN ID for all links.

RAPS Message Termination

The RAPS message starts from the originating node, travels through the entire ring, and terminates in the originating node unless a failure is present in the ring. The originating node must drop the RAPS message if the source MAC address in the RAPS message belongs to itself. The source MAC address is the node's node ID.

Multiple Rings

The Ethernet ring control module supports multiple rings in each node (two logical interfaces are part of each ring). However, interconnection of multiple rings is not supported in this release. The interconnection of two rings means that two rings may share the same link or share the same node.

Node ID

For each node in the ring, a unique *node ID* identifies each node. The node ID is the node's MAC address.

For routers only, you can configure this node ID when configuring the ring on the node or automatically select an ID such as STP. In most cases, you will not configure this and the router will select a node ID, like STP does. It should be the manufacturing MAC address. The ring node ID should not be changed, even if you change the manufacturing MAC address. Any MAC address can be used if you make sure each node in the ring has a different node ID. The node ID on switches is selected automatically and is not configurable.

Bridge Domains with the Ring Port (MX Series Routers Only)

On the routers, the protection group is seen as an abstract logical port that can be configured to any bridge domain. Therefore, if you configure one ring port or its logical interface in a bridge domain, you must configure the other related ring port or its logical interface to the same bridge domain. The bridge domain that includes the ring port acts as any other bridge domain and supports the IRB Layer 3 interface.

- Related Documentation**
- [Ethernet Ring Protection Switching Overview on page 3](#)
 - [Configuring Ethernet Ring Protection Switching](#)

- *Example: Ethernet Ring Protection Switching Configuration on MX Routers*
- *Ethernet Interfaces Feature Guide for Routing Devices*
- *Example: Configuring Ethernet Ring Protection Switching on EX Series Switches*
- [Configuring Ethernet Ring Protection Switching \(CLI Procedure\) on page 9](#)

Configuring Ethernet Ring Protection Switching (CLI Procedure)

You can configure Ethernet ring protection switching (ERPS) on connected switches to prevent fatal loops from disrupting a network. ERPS is similar to spanning-tree protocols, but ERPS is more efficient than spanning-tree protocols because it is customized for ring topologies. You must configure at least three switches to form a ring. One of the links, called the ring protection link (RPL) end interface, is blocked until another link fails—at this time the RPL link is unblocked, ensuring connectivity.



NOTE: Ethernet OAM connectivity fault management (CFM) can be used with ERPS to detect link faults faster in some cases. See *Configuring Ethernet OAM Connectivity Fault Management (CLI Procedure)*.

The time needed for switchover to the ERPS link is affected by three settings—link failure detection time, the number of nodes in the ring, and the time it takes to unblock the RPL after a failure is detected.



NOTE: Do not configure redundant trunk groups on ERPS interfaces. You can configure VSTP on ERPS interfaces if the VSTP uses a VLAN that is not part of the ERPS control VLAN or data channel VLANs. The total number of ERPS and VSTP or MSTP instances is limited to 253.

Before you begin:

- Configure a VLAN to act as a control channel for ERPS. Two interfaces (east and west) on each switch in the ring must be associated with the control VLAN. See *Configuring VLANs for EX Series Switches (CLI Procedure)*.
 - The interfaces on the ERPS control channel are usually (but not required to be) configured as trunk ports. See *Configuring Gigabit Ethernet Interfaces (CLI Procedure)*. Note that if one switch has trunk ports as the ERPS control interfaces, the same must be true of all switches on the ring (the ERPS control interfaces must also be trunk ports).
- Data channels are optional on the ERPS link. If you plan to use them, configure a VLAN for each data channel. If you have multiple ERPS instances, the control VLANs and data channel VLANs must not overlap.

To configure ERPS:



NOTE: You must configure at least three switches, with only one switch designated as the RPL owner node.

1. Spanning tree protocols and ERPS cannot both be configured on the ring ports, so on each ERPS interface, you must disable any configured spanning tree protocols (such as STP, RSTP, VSTP, or MSTP). Spanning tree protocols are disabled for individual interfaces in two different ways, depending on which Junos OS version and release is running on the switch. RSTP is enabled in the default configuration, so disabling RSTP is shown here.

For switches without Enhanced Layer 2 (ELS) software support, and switches running ELS software with Junos OS release 15.1 or later, use this command to disable RSTP on the individual ERPS interfaces:

```
[edit protocols]
user@switch# set rstp interface interface-name disable
```

For switches running Enhanced Layer 2 (ELS) software with Junos OS releases prior to 15.1, you disable spanning tree protocols on individual interfaces by deleting that configuration item. Use this command to delete the RSTP configuration item on the individual ERPS interfaces:

```
[edit protocols]
user@switch# delete rstp interface interface-name
```

2. Create a node ring on each switch:

```
[edit protocols]
user@switch# set protection-group ethernet-ring ring-name
```

3. Configure a control VLAN for the node ring:

```
[edit protocols protection-group ethernet-ring ring name]
user@switch# set control-vlan vlan-name-or-vlan-id
```

4. Configure the east and west interfaces of the node ring with the control-channel interface.

```
[edit protocols protection-group ethernet-ring ring-name]
user@switch# set east-interface control-channel control-channel-name
user@switch# set west-interface control-channel control-channel-name
```

For switches with ELS support, additionally associate the east and west interfaces with the control VLAN:

```
[edit protocols protection-group ethernet-ring ring-name]
user@switch# set east-interface control-channel vlan vlan-name-or-vlan-id
user@switch# set west-interface control-channel vlan vlan-name-or-vlan-id
```

5. In addition, configure either the east interface or the west interface (but not both) as a link end. For example, configure the east interface:

```
[edit protocols protection-group ethernet-ring ring-name]
user@switch# set east-interface ring-protection-link-end
```


6. Configure only one switch as the RPL owner node:

```
[edit protocols protection-group ethernet-ring ring-name]
user@switch# set ring-protection-link-owner
```

7. The restore interval is the time the RPL owner node waits after the last ring automatic protection switching (RAPS) signal failure (SF) event has been cleared, to see if any further RAPS events occur. During this time interval, the RPL owner continues to process RAPS packets, and the ring remains in protection state with the RPL link unblocked. When this interval expires, if no further RAPS SF events have been reported, the RPL owner reverts the protection switching, blocks the RPL link, and returns the protection ring to idle state. Optionally, configure a local restore interval for the ERPS ring on each switch:

```
[edit protocols protection-group ethernet-ring ring-name]
user@switch# set restore-interval restore-interval-value
```



NOTE: The restore interval can also be set globally to apply to any ERPS rings configured on the switch. Local per-ring settings take priority over global settings.

8. The guard interval prevents ring nodes from receiving outdated RAPS messages. Optionally, configure the guard interval on each switch:

```
[edit protocols protection-group ethernet-ring ring name]
user@switch# set guard-interval guard-interval-value
```



NOTE: The guard interval can also be set globally to apply to any ERPS rings configured on the switch. Local per-ring settings take priority over global settings.

9. Global restore and guard interval settings are used when no local settings are configured. If these intervals are not configured globally or locally, the default values apply. Optionally configure global interval settings on the switch to apply to all rings that do not have a corresponding interval configured locally for the ring:

- restore interval:

```
[edit protocols protection-group]
user@switch# set restore-interval restore-interval-value
```

- guard interval:

```
[edit protocols protection-group]
user@switch# set guard-interval guard-interval-value
```



NOTE: You can also configure other global settings, such as ERP traceoptions (file, page size, file size, flag name).

10. Optionally, configure VLANs for data channels on the ERPS link:

```
[edit protocols protection-group ethernet-ring ring name]  
user@switch# set data-channel vlan-name
```

**Related
Documentation**

- [Example: Configuring Ethernet Ring Protection Switching on EX Series Switches](#)
- [Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12](#)
- [Ethernet Ring Protection Switching Overview on page 3](#)
- [Understanding Ethernet Ring Protection Switching Functionality on page 4](#)

Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS

You can configure Ethernet ring protection switching (ERPS) on connected EX Series or QFX Series switches to prevent fatal loops from disrupting a network. ERPS is similar to the Spanning Tree Protocol, but ERPS is more efficient because it is customized for ring topologies. You must connect and configure at least three switches to form a ring.

This example shows how to configure Ethernet ring protection switching on four switches with ELS support, connected to one another on a dedicated link in a ring topology. You can include different types of switches in an ERPS ring, including those with and without ELS support. If any of your EX Series switches runs software that does not support ELS, use these configuration directions: *Example: Configuring Ethernet Ring Protection Switching on EX Series Switches*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

- [Requirements on page 12](#)
- [Overview and Topology on page 13](#)
- [Configuration on page 14](#)
- [Verification on page 26](#)

Requirements

This example uses the following hardware and software components:

- Four connected EX Series switches or QFX Series switches that support the Enhanced Layer 2 Software (ELS) to function as nodes in the ring topology. You could use any of these QFX Series switches: QFX5100, QFX5200, and QFX10000. This configuration also applies to EX Series switches that support the Enhanced Layer 2 Software (ELS) configuration style that runs on EX4300 and EX4600 switches.
- Junos OS Release 13.2X50-D10 or later for EX Series switches.
- Junos OS Release 14.1X53-D10 or later for QFX5100 switches. Junos OS Release 15.1X53-D30 or later for QFX5200, and QFX10000 switches.

Before you begin, be sure you have:

- Configured two trunk interfaces on each of the four switches. See [Table 3 on page 14](#) for a list of the interface names used in this example.

- Configured a VLAN (with name **erp-control-vlan-1** and ID **100**) on all four switches and associated two network interfaces from each of the four switches with the VLAN. See *Configuring VLANs for the QFX Series OR Configuring VLANs for EX Series Switches (CLI Procedure)*. See [Table 3 on page 14](#) for a list of the interface names used in this example.
- Configured two more VLANs (one with name **erp-data-1** and vlan ID **101** and a second vlan with the name **erp-data-2** and vlan ID **102**) on all four switches and associated both the east and west interfaces on each switch.

Overview and Topology

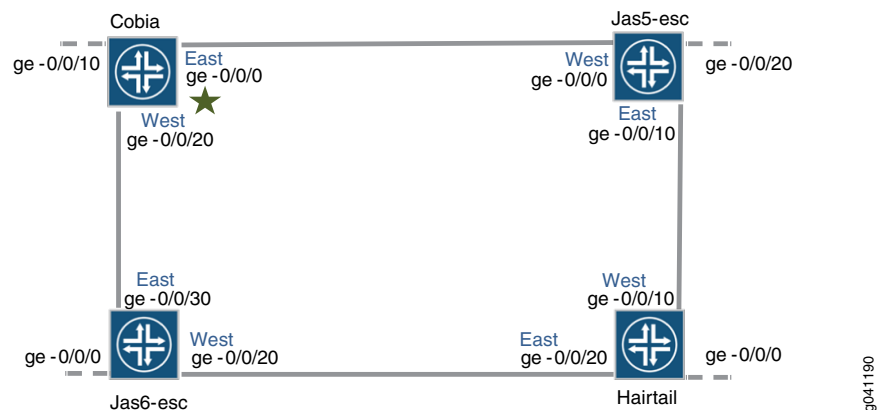
ERPS uses a dedicated physical link, including a control VLAN for trunk ports, between all of the switches to protect the active links. ERPS VLANs are all located on this link and are also blocked by default. When traffic between the switches is flowing with no problems, the active links take care of all traffic. Only if an error occurs on one of the data links would the ERPS control channel take over and start forwarding traffic.



NOTE: Trunk ports on switches use a VLAN to create individual control channels for ERPS. When multiple ERPS instances are configured for a ring, there are multiple sets of ring protection links (RPLs) and RPL owners on the ERPS link, and a different channel is blocked for each instance. Nontrunk ports use the physical link as the control channel and protocol data units (PDUs) are untagged, with no VLAN information in the packet.

This example creates one protection ring (called a node ring) named **erp1** on four switches connected in a ring by trunk ports as shown in [Figure 3 on page 13](#). Because the links are trunk ports, VLAN 100 is used for **erp1** traffic. The east interface of each switch is connected with the west interface of an adjacent switch. Cobia is the RPL owner, with interface **ge-0/0/0** configured as an RPL end interface. The interface **ge-0/0/0** of **Jas5-esc** is configured as the RPL neighbor interface. In the idle state, the RPL end blocks the control VLAN and data channel VLAN for this particular ERP instance—the blocked port on Cobia is marked with a star in [Figure 3 on page 13](#).

Figure 3: Ethernet Ring Protection Switching Example



In this example, we configure the four switches with the interfaces indicated in both [Figure 3 on page 13](#) and [Table 3 on page 14](#).

Table 3: Components to Configure for This Example

Interfaces	Cobia	Jas5-esc	Jas6-esc	Hairtail
East	ge-0/0/0	ge-0/0/10	ge-0/0/30	ge-0/0/20
West	ge-0/0/20	ge-0/0/0	ge-0/0/20	ge-0/0/10
Third	ge-0/0/10	ge-0/0/20	ge-0/0/0	ge-0/0/0

Configuration

- [Configuring ERPS on Cobia, the RPL Owner Node on page 14](#)
- [Configuring ERPS on Jas5-esc on page 17](#)
- [Configuring ERPS on Hairtail on page 20](#)
- [Configuring ERPS on Jas6-esc on page 23](#)

Configuring ERPS on Cobia, the RPL Owner Node

CLI Quick Configuration

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

ERPS cannot be configured on an interface if any spanning-tree protocol is configured. (RSTP is configured by default.) Therefore, in this example, RSTP is disabled on each ring port before configuring ERPS. Spanning-tree protocols are disabled two different ways, depending on which version of the Junos OS you are running. Therefore, the first two statements in this example vary: Junos release 15.1 or later uses one command to turn off RSTP and Junos releases prior to 15.1 uses another command.

```

Junos OS release 15.1 or later: set protocols rstp interface ge-0/0/0 disable
Junos OS release 15.1 or later: set protocols rstp interface ge-0/0/20 disable
Junos OS release prior to 15.1: delete rstp interface ge-0/0/0
Junos OS release prior to 15.1: delete rstp interface ge-0/0/20
set protocols protection-group ethernet-ring erp1
set protocols protection-group ethernet-ring erp1 ring-protection-link-owner
set protocols protection-group ethernet-ring erp1 data-channel 101
set protocols protection-group ethernet-ring erp1 data-channel 102
set protocols protection-group ethernet-ring erp1 control-vlan 100
set protocols protection-group ethernet-ring erp1 east-interface control-channel
  ge-0/0/0.0
set protocols protection-group ethernet-ring erp1 east-interface ring-protection-link-end
set protocols protection-group ethernet-ring erp1 east-interface control-channel
  ge-0/0/20.0
set protocols protection-group ethernet-ring erp1 west-interface control-channel vlan
  100
set protocols protection-group ethernet-ring erp1 east-interface control-channel vlan
  100

```

**Step-by-Step
Procedure**

To configure ERPS on Cobia:

1. Disable any spanning-tree protocol currently configured on the ERPS interfaces. RSTP, VSTP, and MSTP are all available spanning-tree protocols. RSTP is enabled in the default configuration, so this example shows disabling RSTP. Spanning-tree protocols are disabled two different ways, depending on which version of the Junos OS you are running.

If you are running Junos release 15.1 or later, disable any spanning-tree protocol with these commands. To disable RSTP:

```
[edit protocols]
user@switch# set rstp interface ge-0/0/0 disable
user@switch# set rstp interface ge-0/0/20 disable
```

If you are running a Junos release prior to 15.1, disable any spanning-tree protocol with these commands. To disable RSTP:

```
[edit protocols]
user@switch# delete rstp interface ge-0/0/0
user@switch# delete rstp interface ge-0/0/20
```

2. Create a node ring named erp1:

```
[edit protocols]
user@switch# set protection-group ethernet-ring erp1
```

3. Designate Cobia as the RPL owner node:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set ring-protection-link-owner
```

4. Configure the VLANs 101 and 102 as data channels:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set data-channel 101
user@switch# set data-channel 102
```

5. Configure the control vlan 100 for this ERP instance on the trunk interface:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set control-vlan 100
```

6. Configure the east interface of the node ring erp1 with control channel ge-0/0/0.0 and indicate that this particular ring protection link ends here:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set east-interface control-channel ge-0/0/0.0
user@switch# set east-interface ring-protection-link-end
```

7. Configure the west interface of the node ring erp1 with control channel ge-0/0/20.0 :

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set west-interface control-channel ge-0/0/20.0
```

8. Every ring instance on a trunk port has one control VLAN in which ERP packets traverse. The control VLAN also controls data VLANs, if any are configured. Assign 100 as the control VLAN on both interfaces:

```
[edit protocols protection-group ethernet-ring erp1]
```

```

user@switch# set west-interface control-channel vlan 100
user@switch# set east-interface control-channel vlan 100

```

Results In configuration mode, check your ERPS configuration by entering the **show configuration protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```

[edit]
user@switch# show configuration protocols
  protection-group {
    ethernet-ring ERP1 {
      restore-interval 5;
      east-interface {
        control-channel {
          ge-0/0/13.0;
          vlan 4000;
        }
      }
      west-interface {
        control-channel {
          ge-0/0/15.0;
          vlan 4000;
        }
      }
      control-vlan 4000;
      data-channel {
        vlan 2052;
      }
    }
  }
}
user@switch# show configuration protocols | display set
set protocols protection-group ethernet-ring ERP1 restore-interval 5
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
ge-0/0/13.0
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
ge-0/0/15.0
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 control-vlan 4000
set protocols protection-group ethernet-ring ERP1 data-channel vlan 2052

```

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

In configuration mode, check your interface configurations by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```

[edit]
show vlans
  erp-control-vlan-1 {
    vlan-id 100;
  }
  erp-data-1 {
    vlan-id 101;
  }

```

```

    }
    erp-data-2 {
        vlan-id 102;
    }
}
user@switch# show interfaces
ge-0/0/0 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members [ 100 101 102 ];
            }
        }
    }
}
ge-0/0/10 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members [ 101 102 ];
            }
        }
    }
}
ge-0/0/20 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members [ 100 101 102 ];
            }
        }
    }
}
}

```

If you are finished configuring the device, enter **commit** in configuration mode.

Configuring ERPS on Jas5-esc

CLI Quick Configuration

To quickly configure Jas5-esc, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

ERPS cannot be configured on an interface if any spanning tree protocol is configured. (RSTP is configured by default.) Therefore, in this example, RSTP is disabled on each ring port before configuring ERPS. Spanning tree is disabled two different ways, depending on which version of the Junos OS you are running. Therefore, the first two statements will vary: Junos release 15.1 or later uses one command to turn off RSTP and Junos releases prior to 15.1 uses another command.

```

Junos OS release 15.1 or later: set protocols rstp interface ge-0/0/10 disable
Junos OS release 15.1 or later: set protocols rstp interface ge-0/0/0 disable
Junos OS release prior to 15.1: delete rstp interface ge-0/0/10
Junos OS release prior to 15.1: delete rstp interface ge-0/0/0
set protocols protection-group ethernet-ring erp1
set protocols protection-group ethernet-ring erp1 data-channel 101
set protocols protection-group ethernet-ring erp1 data-channel 102

```

```

set protocols protection-group ethernet-ring erp1 control-vlan 100
set protocols protection-group ethernet-ring erp1 east-interface control-channel
  ge-0/0/0.0
set protocols protection-group ethernet-ring erp1 west-interface control-channel
  ge-0/0/20.0
set protocols protection-group ethernet-ring erp1 west-interface control-channel vlan
  100
set protocols protection-group ethernet-ring erp1 west-interface control-channel vlan
  100

```

Step-by-Step Procedure

To configure ERPS on Jas5-esc:

1. Disable any spanning-tree protocol currently configured on the ERPS interfaces. RSTP, VSTP, and MSTP are all available spanning-tree protocols. RSTP is enabled in the default configuration, so this example shows disabling RSTP. Spanning-tree protocols are disabled two different ways, depending on which version of the Junos OS you are running.

If you are running Junos release 15.1 or later, disable any spanning-tree protocol with these commands. To disable RSTP:

```

[edit protocols]
user@switch# set rstp interface ge-0/0/10 disable
user@switch# set rstp interface ge-0/0/0 disable

```

If you are running a Junos release prior to 15.1, disable any version of spanning-tree protocol with these commands. To disable RSTP:

```

[edit protocols]
user@switch# delete rstp interface ge-0/0/10
user@switch# delete rstp interface ge-0/0/0

```

2. Create a node ring named erp1:
3. Configure two data channels named erp-data-1 and erp-data-2 to define a set of VLAN IDs that belong to a ring instance.

```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set data-channel vlan 101
user@switch# set data-channel vlan 102

```

4. Configure a control VLAN with ID 100 for the node ring erp1:

```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set control-vlan 100

```

5. Configure the east interface of the node ring erp1 with the control channel ge-0/0/10.0:

```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set east-interface control-channel ge-0/0/10.0

```

6. Configure the west interface of the node ring erp1 with the control channel ge-0/0/0.0 vlan 100:

```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set west-interface control-channel ge-0/0/0.0

```


7. Every ring instance on a trunk port has one control VLAN in which ERP packets traverse. The control VLAN also controls data VLANs, if any are configured. Assign vlan # 100 as the control VLAN:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set west-interface control-channel vlan 100
user@switch# set east-interface control-channel vlan 100
```

Results In configuration mode, check your ERPS configuration by entering the **show configuration protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@switch# show configuration protocols
  protection-group {
    ethernet-ring ERP1 {
      restore-interval 5;
      east-interface {
        control-channel {
          ge-0/0/13.0;
          vlan 4000;
        }
      }
      west-interface {
        control-channel {
          ge-0/0/15.0;
          vlan 4000;
        }
      }
      control-vlan 4000;
      data-channel {
        vlan 2052;
      }
    }
  }
}

user@switch# show configuration protocols | display set
set protocols protection-group ethernet-ring ERP1 restore-interval 5
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
ge-0/0/13.0
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
ge-0/0/15.0
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 control-vlan 4000
set protocols protection-group ethernet-ring ERP1 data-channel vlan 2052
```

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

In configuration mode, check your interface configurations by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
show vlans
```

```

    erp-control-vlan-1 {
        vlan-id 100;
    }
    erp-data-1 {
        vlan-id 101;
    }
    erp-data-2 {
        vlan-id 102;
    }
}
user@switch# show interfaces
ge-0/0/0 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members [ 100 101 102 ];
            }
        }
    }
}
ge-0/0/10 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members [ 100 101 102 ];
            }
        }
    }
}
ge-0/0/20 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members [ 101 102 ];
            }
        }
    }
}

```

If you are finished configuring the device, enter **commit** in configuration mode.

Configuring ERPS on Hairtail

CLI Quick Configuration

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

ERPS cannot be configured on an interface if any spanning tree protocol is configured. (RSTP is configured by default.) Therefore, in this example, RSTP is disabled on each ring port before configuring ERPS. Spanning tree is disabled two different ways, depending on which version of the Junos OS you are running. Therefore, the first two statements will vary: Junos release 15.1 or later uses one command to turn off RSTP and Junos releases prior to 15.1 uses another command.

Junos OS release 15.1 or later: **set protocols rstp interface ge-0/0/10 disable**

```

Junos OS release 15.1 or later: set protocols rstp interface ge-0/0/20 disable
Junos OS release prior to 15.1: delete rstp interface ge-0/0/10
Junos OS release prior to 15.1: delete rstp interface ge-0/0/20
set protocols protection-group ethernet-ring erp1
set protocols protection-group ethernet-ring erp1 data-channel 101
set protocols protection-group ethernet-ring erp1 data-channel 102
set protocols protection-group ethernet-ring erp1 control-vlan 100
Set protocols protection-group ethernet-ring erp1 east-interface control-channel
ge-0/0/0.0
set protocols protection-group ethernet-ring erp1 east-interface control-channel
ge-0/0/20.0
set protocols protection-group ethernet-ring erp1 west-interface control-channel vlan
100
set protocols protection-group ethernet-ring erp1 east-interface control-channel vlan
100

```

Step-by-Step Procedure

To configure ERPS on Hairtail:

1. Disable any spanning-tree protocol currently configured on the ERPS interfaces. RSTP, VSTP, and MSTP are all available spanning-tree protocols. RSTP is enabled in the default configuration, so this example shows disabling RSTP. Spanning-tree protocols are disabled two different ways, depending on which version of the Junos OS you are running.

If you are running Junos release 15.1 or later, disable any spanning-tree protocol with these commands. To disable RSTP:

```

[edit protocols]
user@switch# set rstp interface ge-0/0/10 disable
user@switch# set rstp interface ge-0/0/20 disable

```

If you are running a Junos release prior to 15.1, disable any spanning-tree protocol with these commands. To disable RSTP:

```

[edit protocols]
user@switch# delete rstp interface ge-0/0/10
user@switch# delete rstp interface ge-0/0/20

```

2. Create a node ring named erp1:


```

[edit protocols]
user@switch# set protection-group ethernet-ring erp1

```
3. Configure the control vlan 100 for the node ring erp1:


```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set control-vlan 100

```
4. Configure two data channels numbered 101 and 102 to define a set of VLAN IDs that belong to a ring instance:


```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set data-channel vlan 101
user@switch# set data-channel vlan 102

```
5. Configure the east interface of the node ring erp1 with the control channel ge-0/0/20.0:


```

[edit protocols protection-group ethernet-ring erp1]

```

```
user@switch# set east-interface control-channel ge-0/0/20.0
```

6. Configure the west interface of the node ring erp1 with the control channel ge-0/0/10.0:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set west-interface control-channel ge-0/0/10.0
```

7. Every ring instance on a trunk port has one control VLAN in which ERP packets traverse. The control VLAN also controls data VLANs, if any are configured. Assign 100 as the control VLAN:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set west-interface control-channel vlan 100
user@switch# set east-interface control-channel vlan 100
```

Results In configuration mode, check your ERPS configuration by entering the **show configuration protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@switch# show configuration protocols
  protection-group {
    ethernet-ring ERP1 {
      restore-interval 5;
      east-interface {
        control-channel {
          ge-0/0/13.0;
          vlan 4000;
        }
      }
      west-interface {
        control-channel {
          ge-0/0/15.0;
          vlan 4000;
        }
      }
      control-vlan 4000;
      data-channel {
        vlan 2052;
      }
    }
  }
}
user@switch# show configuration protocols | display set
set protocols protection-group ethernet-ring ERP1 restore-interval 5
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
ge-0/0/13.0
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
ge-0/0/15.0
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 control-vlan 4000
set protocols protection-group ethernet-ring ERP1 data-channel vlan 2052
```

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

In configuration mode, check your interface configurations by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
show vlans
  erp-control-vlan-1 {
    vlan-id 100;
  }
  erp-data-1 {
    vlan-id 101;
  }
  erp-data-2 {
    vlan-id 102;
  }
user@switch# show interfaces
ge-0/0/0 {
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members [ 101 102 ];
      }
    }
  }
}
ge-0/0/10 {
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members [ 100 101 102 ];
      }
    }
  }
}
ge-0/0/20 {
  unit 0 {
```

If you are finished configuring the device, enter **commit** in configuration mode.

Configuring ERPS on Jas6-esc

CLI Quick Configuration

To quickly configure Jas6-esc, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

ERPS cannot be configured on an interface if any spanning tree protocol is configured. (RSTP is configured by default.) Therefore, in this example, RSTP is disabled on each ring port before configuring ERPS. Spanning tree is disabled two different ways, depending on which version of the Junos OS you are running. Therefore, the first two statements will vary: Junos release 15.1 or later uses one command to turn off RSTP and Junos releases prior to 15.1 uses another command.

```
Junos OS release 15.1 or later: set protocols rstp interface ge-0/0/30 disable
Junos OS release 15.1 or later: set protocols rstp interface ge-0/0/20 disable
```

```

Junos OS release prior to 15.1: delete rstp interface ge-0/0/30
Junos OS release prior to 15.1: delete rstp interface ge-0/0/20
set protocols protection-group ethernet-ring erp1
set protocols protection-group ethernet-ring erp1 data-channel 101
set protocols protection-group ethernet-ring erp1 data-channel 102
set protocols protection-group ethernet-ring erp1 control-vlan 100
set protocols protection-group ethernet-ring erp1 east-interface control-channel
  ge-0/0/30.0
set protocols protection-group ethernet-ring erp1 west-interface control-channel
  ge-0/0/20.0
set protocols protection-group ethernet-ring erp1 west-interface control-channel vlan
  100
set protocols protection-group ethernet-ring erp1 east-interface control-channel vlan
  100

```

Step-by-Step Procedure

To configure ERPS on Jas6-esc:

1. Disable any spanning-tree protocol currently configured on the ERPS interfaces. RSTP, VSTP, and MSTP are all available spanning-tree protocols. RSTP is enabled in the default configuration, so this example shows disabling RSTP. Spanning-tree protocols are disabled two different ways, depending on which version of the Junos OS you are running.

If you are running Junos release 15.1 or later, disable any spanning-tree protocol with these commands. To disable RSTP:

```

[edit protocols]
user@switch# set rstp interface ge-0/0/30 disable
user@switch# set rstp interface ge-0/0/20 disable

```

If you are running a Junos release prior to 15.1, disable any spanning-tree protocol with these commands. To disable RSTP:

```

[edit protocols]
user@switch# delete rstp interface ge-0/0/30
user@switch# delete rstp interface ge-0/0/20

```

2. Create a node ring named erp1:


```

[edit protocols]
user@switch# set protection-group ethernet-ring erp1

```
3. Configure the control vlan 100 for the node ring erp1:


```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set control-vlan 100

```
4. Configure two data channels numbered 101 and 102 to define VLAN IDs that belong to a ring instance.


```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set data-channel 101
user@switch# set data-channel 102

```
5. Configure the east interface of the node ring erp1 with the control channel ge-0/0/30.0 :


```

[edit protocols protection-group ethernet-ring erp1]
user@switch# set east-interface control-channel ge-0/0/30.0

```

6. Configure the west interface of the node ring erp1 with the control channel ge-0/0/20.0:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set west-interface control-channel ge-0/0/20.0
```
7. Every ring instance on a trunk port has one control VLAN in which ERP packets traverse. The control VLAN also controls data VLANs, if any are configured. Assign vlan number 100 as the control VLAN:

```
[edit protocols protection-group ethernet-ring erp1]
user@switch# set west-interface control-channel vlan 100
user@switch# set east-interface control-channel vlan 100
```

Results In configuration mode, check your ERPS configuration by entering the **show configuration protocols** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@switch# show configuration protocols
}
  protection-group {
    ethernet-ring ERP1 {
      restore-interval 5;
      east-interface {
        control-channel {
          ge-0/0/13.0;
          vlan 4000;
        }
      }
      west-interface {
        control-channel {
          ge-0/0/15.0;
          vlan 4000;
        }
      }
      control-vlan 4000;
      data-channel {
        vlan 2052;
      }
    }
  }
}

user@switch# show configuration protocols | display set
set protocols protection-group ethernet-ring ERP1 restore-interval 5
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
ge-0/0/13.0
set protocols protection-group ethernet-ring ERP1 east-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
ge-0/0/15.0
set protocols protection-group ethernet-ring ERP1 west-interface control-channel
vlan 4000
set protocols protection-group ethernet-ring ERP1 control-vlan 4000
set protocols protection-group ethernet-ring ERP1 data-channel vlan 2052
```

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

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

```
[edit]
show vlans
  erp-control-vlan-1 {
    vlan-id 100;
  }
  erp-data-1 {
    vlan-id 101;
  }
  erp-data-2 {
    vlan-id 102;
  }
user@switch# show interfaces
ge-0/0/0 {
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members [ 101 102 ];
      }
    }
  }
}
ge-0/0/20 {
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members [ 100 101 102 ];
      }
    }
  }
}
ge-0/0/30 {
  unit 0 {
    family ethernet-switching {
      interface-mode trunk;
      vlan {
        members [ 100 101 102 ];
      }
    }
  }
}
```

Verification

Verify that ERPS is working correctly.

Verifying That ERPS Is Working Correctly

Purpose Verify that ERPS is working on the four EX switches that function as nodes in the ring topology.

Action Check the state of the ring links in the output of the **show protection-group ethernet-ring interface** command. When the ring is configured but not being used (no error exists on

the data links), one ERP interface is forwarding traffic and one is discarding traffic. Discarding blocks the ring.

```
user@switch> show protection-group ethernet-ring interface
```

```
Ethernet ring port parameters for protection group ERP1
```

Interface	Control Channel	Forward State	Ring Protection	Link End	Signal
ge-0/0/13	ge-0/0/13.0	forwarding	No		Clear
	IFF ready				
ge-0/0/15	ge-0/0/15.0	forwarding	No		Clear
	IFF ready				

To find out what has occurred since the last restart, check the RPS statistics for ring-blocked events. **NR** is a No Request ring block, which means that the switch is not blocking either of the two ERP interfaces. **NR-RB** is a No Request Ring Blocked event, which means that the switch is blocking one of its ERP interfaces and sending a packet out to notify the other switches.

```
user@switch> show protection-group ethernet-ring statistics
```

```
Ethernet Ring statistics for PG erp1
RAPS event sent           : 0
RAPS event received       : 1
Local SF happened:        : 0
Remote SF happened:       : 0
NR event happened:        : 0
NR-RB event happened:     : 1
```

Meaning The **show protection-group ethernet-ring interface** command output from the RPL owner node indicates that one interface is forwarding traffic and one is discarding traffic, meaning that the ERP is ready but not active. If at least one interface in the ring is not forwarding, the ring is blocked and therefore ERP is working.

The **show protection-group ethernet-ring statistics** command output indicates that, since the last reboot, both local and remote signal failures have occurred (**Local SF** and **Remote SF**).

The **NR Event** count is 2, indicating that the NR state was entered into twice. **NR** stands for No Request. This means that the switch either originated NR PDUs or received an NR PDU from another switch and stopped blocking the interface to allow ERP to function.

The three **NR-RB** events indicate that on three occasions, this switch either sent out NR-RB PDUs or received NR-RB PDUs from another switch. This occurs when a network problem is resolved and the switch once again blocks the ERP link at one end.

Related Documentation

- [Configuring Ethernet Ring Protection Switching \(CLI Procedure\) on page 9](#)
- [Ethernet Ring Protection Switching Overview on page 3](#)
- [Understanding Ethernet Ring Protection Switching Functionality on page 4](#)

PART 2

Bridging and VLANs

- [Using Bridging and VLANs on page 31](#)

CHAPTER 2

Using Bridging and VLANs

- [Overview of Layer 2 Networking on page 31](#)
- [Understanding Layer 2 Broadcasting on page 34](#)
- [Layer 2 Learning and Forwarding for VLANs Overview on page 34](#)
- [Understanding Bridging and VLANs on page 35](#)
- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Example: Setting Up Bridging with Multiple VLANs on page 60](#)
- [Configuring VLANs on page 66](#)
- [Configuring the Native VLAN Identifier \(CLI Procedure\) on page 67](#)
- [Creating a Series of Tagged VLANs on page 69](#)
- [Understanding Integrated Routing and Bridging on page 70](#)
- [Configuring IRB Interfaces on page 72](#)
- [Example: Configuring Routing Between VLANs on One Switch on page 73](#)
- [Excluding a Routed VLAN Interface from State Calculations on page 79](#)
- [Adding a Static MAC Address Entry to the Ethernet Switching Table \(CLI Procedure\) on page 81](#)
- [Configuring Static ARP Entries on page 81](#)
- [Understanding Multiple VLAN Registration Protocol \(MVRP\) on page 82](#)
- [Troubleshooting Ethernet Switching on page 84](#)

Overview of Layer 2 Networking

Layer 2, also known as the Data Link Layer, is the second level in the seven-layer OSI reference model for network protocol design. Layer 2 is equivalent to the link layer (the lowest layer) in the TCP/IP network model. Layer 2 is the network layer used to transfer data between adjacent network nodes in a wide area network or between nodes on the same local area network.

A *frame* is a protocol data unit, the smallest unit of bits on a Layer 2 network. Frames are transmitted to and received from devices on the same local area network (LAN). Unlike bits, frames have a defined structure and can be used for error detection, control plane activities and so forth. Not all frames carry user data. The network uses some frames to control the data link itself..

At Layer 2, *unicast* refers to sending frames from one node to a single other node, whereas *multicast* denotes sending traffic from one node to multiple nodes, and *broadcasting* refers to the transmission of frames to all nodes in a network. A *broadcast domain* is a logical division of a network in which all nodes of that network can be reached at Layer 2 by a broadcast.

Segments of a LAN can be linked at the frame level using *bridges*. Bridging creates separate broadcast domains on the LAN, creating VLANs, which are independent logical networks that group together related devices into separate network segments. The grouping of devices on a VLAN is independent of where the devices are physically located in the LAN. Without bridging and VLANs, all devices on the Ethernet LAN are in a single broadcast domain, and all the devices detect all the packets on the LAN.

Forwarding is the relaying of packets from one network segment to another by nodes in the network. On a VLAN, a frame whose origin and destination are in the same VLAN are forwarded only within the local VLAN. A network segment is a portion of a computer network wherein every device communicates using the same physical layer.

Layer 2 contains two sublayers:

- Logical link control (LLC) sublayer, which is responsible for managing communications links and handling frame traffic.
- Media access control (MAC) sublayer, which governs protocol access to the physical network medium. By using the MAC addresses that are assigned to all ports on a switch, multiple devices on the same physical link can uniquely identify one another.

The ports, or interfaces, on a switch operate in either access mode, tagged-access, or trunk mode:

- *Access mode* ports connect to a network device such as a desktop computer, an IP telephone, a printer, a file server, or a security camera. The port itself belongs to a single VLAN. The frames transmitted over an access interface are normal Ethernet frames. By default, all ports on a switch are in access mode.
- *Tagged-Access mode* ports connect to a network device such as a desktop computer, an IP telephone, a printer, a file server, or a security camera. The port itself belongs to a single VLAN. The frames transmitted over an access interface are normal Ethernet frames. By default, all ports on a switch are in access mode. Tagged-access mode accommodates cloud computing, specifically scenarios including virtual machines or virtual computers. Because several virtual computers can be included on one physical server, the packets generated by one server can contain an aggregation of VLAN packets from different virtual machines on that server. To accommodate this situation, tagged-access mode reflects packets back to the physical server on the same downstream port when the destination address of the packet was learned on that downstream port. Packets are also reflected back to the physical server on the downstream port when the destination has not yet been learned. Therefore, the third interface mode, tagged access, has some characteristics of access mode and some characteristics of trunk mode:

- *Trunk mode* ports handle traffic for multiple VLANs, multiplexing the traffic for all those VLANs over the same physical connection. Trunk interfaces are generally used to interconnect switches to other devices or switches.

With native VLAN configured, frames that do not carry VLAN tags are sent over the trunk interface. If you have a situation where packets pass from a device to a switch in access mode, and you want to then send those packets from the switch over a trunk port, use native VLAN mode. Configure the single VLAN on the switch's port (which is in access mode) as a native VLAN. The switch's trunk port will then treat those frames differently than the other tagged packets. For example, if a trunk port has three VLANs, 10, 20, and 30, assigned to it with VLAN 10 being the native VLAN, frames on VLAN 10 that leave the trunk port on the other end have no 802.1Q header (tag). There is another native VLAN option. You can have the switch add and remove tags for untagged packets. To do this, you first configure the single VLAN as a native VLAN on a port attached to a device on the edge. Then, assign a VLAN ID tag to the single native VLAN on the port connected to a device. Last, add the VLAN ID to the trunk port. Now, when the switch receives the untagged packet, it adds the ID you specified and sends and receives the tagged packets on the trunk port configured to accept that VLAN.

Including the sublayers, Layer 2 on the QFX Series supports the following functionality:

- Unicast, multicast, and broadcast traffic.
- Bridging.
- VLAN 802.1Q—Also known as *VLAN tagging*, this protocol allows multiple bridged networks to transparently share the same physical network link by adding VLAN tags to an Ethernet frame.
- Extension of Layer 2 VLANs across multiple switches using Spanning Tree Protocol (STP) prevents looping across the network.
- *MAC learning*, including per-VLAN MAC learning and Layer 2 learning suppression—This process obtains the MAC addresses of all the nodes on a network
- Link aggregation—This process groups of Ethernet interfaces at the physical layer to form a single link layer interface, also known as a *link aggregation group (LAG)* or LAG bundle
- Storm control on the physical port for unicast, multicast, and broadcast
- STP support, including 802.1d, RSTP, MSTP, and Root Guard

**Related
Documentation**

- [Understanding Bridging and VLANs on page 35](#)
- *Understanding Bridging and VLANs*

Understanding Layer 2 Broadcasting

In a Layer 2 network, *broadcasting* refers to sending traffic to all nodes on a network.

Layer 2 broadcast traffic stays within a local area network (LAN) boundary; known as the *broadcast domain*. Layer 2 broadcast traffic is sent to the broadcast domain using a MAC address of FF:FF:FF:FF:FF:FF. Every device in the broadcast domain recognizes this MAC address and passes the broadcast traffic on to other devices in the broadcast domain, if applicable. Broadcasting can be compared to unicasting (sending traffic to a single node) or multicasting (delivering traffic to a group of nodes simultaneously).

Layer 3 broadcast traffic, however, is sent to all devices in a network using a broadcast network address. For example, if your network address is 192.0.0.0, the broadcast network address is 192.255.255.255. In this case, only devices that belong to the 192.0.0.0 network receive the Layer 3 broadcast traffic. Devices that do not belong to this network drop the traffic.

Broadcasting is used in the following situations:

- Address Resolution Protocol (ARP) uses broadcasting to map MAC addresses to IP addresses. ARP dynamically binds the IP address (the logical address) to the correct MAC address. Before IP unicast packets can be sent, ARP discovers the MAC address used by the Ethernet interface where the IP address is configured.
- Dynamic Host Configuration Protocol (DHCP) uses broadcasting to dynamically assign IP addresses to hosts on a network segment or subnet.
- Routing protocols use broadcasting to advertise routes.

Excessive broadcast traffic can sometimes create a broadcast storm. A broadcast storm occurs when messages are broadcast on a network and each message prompts a receiving node to respond by broadcasting its own messages on the network. This, in turn, prompts further responses that create a snowball effect. The LAN is suddenly flooded with packets, creating unnecessary traffic that leads to poor network performance or even a complete loss of network service.

Related Documentation

- [Overview of Layer 2 Networking on page 31](#)
- [Understanding Storm Control](#)
- [Understanding Bridging and VLANs](#)
- [Understanding Bridging and VLANs on page 35](#)

Layer 2 Learning and Forwarding for VLANs Overview

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



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

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

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

**Related
Documentation**

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

Understanding Bridging and VLANs

Network switches use Layer 2 bridging protocols to discover the topology of their LAN and to forward traffic toward destinations on the LAN. This topic explains the following concepts regarding bridging and VLANs:

- [History of VLANs on page 35](#)
- [How Bridging of VLAN Traffic Works on page 36](#)
- [Packets Are Either Tagged or Untagged on page 37](#)
- [Switch Interface Modes—Access, Trunk, or Tagged Access on page 37](#)
- [Additional Advantages of Using VLANs on page 39](#)
- [Maximum VLANs and VLAN Members Per Switch on page 40](#)
- [A Default VLAN Is Configured on Most Switches on page 41](#)
- [Assigning Traffic to VLANs on page 41](#)
- [Forwarding VLAN Traffic on page 42](#)
- [VLANs Communicate with Integrated Routing and Bridging Interfaces or Routed VLAN Interfaces on page 42](#)

History of VLANs

Ethernet LANs were originally designed for small, simple networks that primarily carried text. However, over time, the type of data carried by LANs grew to include voice, graphics, and video. This more complex data, when combined with the ever-increasing speed of transmission, eventually became too much of a load for the original Ethernet LAN design. Multiple packet collisions were significantly slowing down the larger LANs.

The IEEE 802.1D-2004 standard helped evolve Ethernet LANs to cope with the higher data and transmission requirements by defining the concept of *transparent bridging* (generally called simply *bridging*). Bridging divides a single physical LAN (now called a single *broadcast domain*) into two or more virtual LANs, or VLANs. Each VLAN is a collection of some of the LAN nodes grouped together to form individual broadcast domains.

When VLANs are grouped logically by function or organization, a significant percentage of data traffic stays within the VLAN. This relieves the load on the LAN because all traffic no longer has to be forwarded to all nodes on the LAN. A VLAN first transmits packets within the VLAN, thereby reducing the number of packets transmitted on the entire LAN. Because packets whose origin and destination are in the same VLAN are forwarded only within the local VLAN, packets that are not destined for the local VLAN are the only ones forwarded to other broadcast domains. This way, bridging and VLANs limit the amount of traffic flowing across the entire LAN by reducing the possible number of collisions and packet retransmissions within VLANs and on the LAN as a whole.

How Bridging of VLAN Traffic Works

Because the objective of the IEEE 802.1D-2004 standard was to reduce traffic and therefore reduce potential transmission collisions for Ethernet, a system was implemented to reuse information. Instead of having a switch go through a location process every time a frame is sent to a node, the transparent bridging protocol allows a switch to record the location of known nodes. When packets are sent to nodes, those destination node locations are stored in address-lookup tables called *Ethernet switching tables*. Before sending a packet, a switch using bridging first consults the switching tables to see if that node has already been located. If the location of a node is known, the frame is sent directly to that node.

Transparent bridging uses five mechanisms to create and maintain Ethernet switching tables on the switch:

- Learning
- Forwarding
- Flooding
- Filtering
- Aging

The key bridging mechanism used by LANs and VLANs is *learning*. When a switch is first connected to an Ethernet LAN or VLAN, it has no information about other nodes on the network. As packets are sent, the switch learns the embedded MAC addresses of the sending nodes and stores them in the Ethernet switching table, along with two other pieces of information—the interface (or port) on which the traffic was received on the destination node and the time the address was learned.

Learning allows switches to then do *forwarding*. By consulting the Ethernet switching table to see whether the table already contains the frame's destination MAC address, switches save time and resources when forwarding packets to the known MAC addresses.

If the Ethernet switching table does not contain an entry for an address, the switch uses flooding to learn that address.

Flooding finds a particular destination MAC address without using the Ethernet switching table. When traffic originates on the switch and the Ethernet switching table does not yet contain the destination MAC address, the switch first floods the traffic to all other interfaces within the VLAN. When the destination node receives the flooded traffic, it can send an acknowledgment packet back to the switch, allowing it to learn the MAC address of the node and add the address to its Ethernet switching table.

Filtering, the fourth bridging mechanism, is how broadcast traffic is limited to the local VLAN whenever possible. As the number of entries in the Ethernet switching table grows, the switch pieces together an increasingly complete picture of the VLAN and the larger LAN—it learns which nodes are in the local VLAN and which are on other network segments. The switch uses this information to filter traffic. Specifically, for traffic whose source and destination MAC addresses are in the local VLAN, filtering prevents the switch from forwarding this traffic to other network segments.

To keep entries in the Ethernet switching table current, the switch uses a fifth bridging mechanism, *aging*. Aging is the reason that the Ethernet switching table entries include timestamps. Each time the switch detects traffic from a MAC address, it updates the timestamp. A timer on the switch periodically checks the timestamp, and if it is older than a user-configured value, the switch removes the node's MAC address from the Ethernet switching table. This aging process eventually flushes unavailable network nodes out of the Ethernet switching table.

Packets Are Either Tagged or Untagged

When an Ethernet LAN is divided into VLANs, each VLAN is identified by a unique 802.1Q ID. The number of available VLANs and VLAN IDs are listed below:

- On a switch running ELS software, you can configure 4093 VLANs.
- On a switch running non-ELS software, you can configure 4091 VLANs.

Ethernet packets include a tag protocol identifier (TPID) EtherType field, which identifies the protocol being transported. When a device within a VLAN generates a packet, this field includes a value of 0x8100, which indicates that the packet is a VLAN-tagged packet. The packet also has a VLAN ID field that includes the unique 802.1Q ID, which identifies the VLAN to which the packet belongs.

In addition to the TPID EtherType value of 0x8100, switches that run Junos OS that does not support the Enhanced Layer 2 Software (ELS) configuration style also support values of 0x88a8 (Provider Bridging and Shortest Path Bridging) and 0x9100 (Q-in-Q).

For a simple network that has only a single VLAN, all packets include a default 802.1Q tag, which is the only VLAN membership that does not mark the packet as tagged. These packets are untagged packets.

Switch Interface Modes—Access, Trunk, or Tagged Access

Ports, or interfaces, on a switch operate in one of three modes:

- Access mode
- Trunk mode
- Tagged-access mode

Access Mode

An interface in access mode connects a switch to a single network device, such as a desktop computer, an IP telephone, a printer, a file server, or a security camera. Access interfaces accept only untagged packets.

By default, when you boot a switch that runs Junos OS that does not support ELS and use the factory default configuration, or when you boot such a switch and do not explicitly configure a port mode, all interfaces on the switch are in access mode and accept only untagged packets from the VLAN named **default**. You can optionally configure another VLAN and use that VLAN instead of **default**.

On a switch that runs Junos OS that supports ELS, the VLAN named **default** is not supported. Therefore, on such switches, you must explicitly configure at least one VLAN, even if your network is simple and you want only one broadcast domain to exist. After you assign an interface to a VLAN, the interface functions in access mode.

For switches that run either type of software, you can also configure a trunk port or interface to accept untagged packets from a user-configured VLAN. For details about this concept (native VLAN), see [“Trunk Mode and Native VLAN” on page 38](#).

Trunk Mode

Trunk mode interfaces are generally used to connect switches to one another. Traffic sent between switches can then consist of packets from multiple VLANs, with those packets multiplexed so that they can be sent over the same physical connection. Trunk interfaces usually accept only tagged packets and use the VLAN ID tag to determine both the packets' VLAN origin and VLAN destination.

On a switch that runs software that does not support ELS, an untagged packet is not recognized on a trunk port unless you configure additional settings on that port.

On a switch that runs Junos OS that supports ELS, a trunk port recognizes untagged control packets for protocols such as the Link Aggregation Control Protocol (LACP) and the Link Layer Discovery Protocol (LLDP). However, the trunk port does not recognize untagged data packets unless you configure additional settings on that port.

In the rare case where you want untagged packets to be recognized by a trunk port on switches that run either type of software, you must configure the single VLAN on a trunk port as a *native VLAN*. For more information about native VLANs, see [“Trunk Mode and Native VLAN” on page 38](#).

Trunk Mode and Native VLAN

On a switch that runs Junos OS that does not support ELS, a trunk port does not recognize packets that do not include VLAN tags, which are also known as untagged packets. On a switch that runs Junos OS that supports ELS, a trunk port recognizes untagged control

packets, but it does not recognize untagged data packets. With native VLAN configured, untagged packets that a trunk port normally does not recognize are sent over the trunk interface. In a situation where packets pass from a device, such as an IP phone or printer, to a switch in access mode, and you want those packets sent from the switch over a trunk port, use native VLAN mode. Create a native VLAN by configuring a VLAN ID for it, and specify that the trunk port is a member of the native VLAN.

The switch's trunk port will then treat those packets differently than the other tagged packets. For example, if a trunk port has three VLANs, 10, 20, and 30, assigned to it with VLAN 10 being the native VLAN, packets on VLAN 10 that leave the trunk port on the other end have no 802.1Q header (tag).

There is another native VLAN option for switches that do not support ELS. You can have the switch add and remove tags for untagged packets. To do this, you first configure the single VLAN as a native VLAN on a port attached to a device on the edge. Then, assign a VLAN ID tag to the single native VLAN on the port connected to a device. Last, add the VLAN ID to the trunk port. Now, when the switch receives the untagged packet, it adds the ID you specified and sends and receives the tagged packets on the trunk port configured to accept that VLAN.

Tagged-Access Mode

Only switches that run Junos OS that does not use the ELS configuration style support tagged-access mode. Tagged-access mode accommodates cloud computing, specifically scenarios including virtual machines or virtual computers. Because several virtual computers can be included on one physical server, the packets generated by one server can contain an aggregation of VLAN packets from different virtual machines on that server. To accommodate this situation, tagged-access mode reflects packets back to the physical server on the same downstream port when the destination address of the packet was learned on that downstream port. Packets are also reflected back to the physical server on the downstream port when the destination has not yet been learned. Therefore, the third interface mode, tagged access, has some characteristics of access mode and some characteristics of trunk mode:

- Like access mode, tagged-access mode connects the switch to an access layer device. Unlike access mode, tagged-access mode is capable of accepting VLAN tagged packets.
- Like trunk mode, tagged-access mode accepts VLAN tagged packets from multiple VLANs. Unlike trunk port interfaces, which are connected at the core/distribution layer, tagged-access port interfaces connect devices at the access layer.

Like trunk mode, tagged-access mode also supports native VLAN.



NOTE: Control packets are never reflected back on the downstream port.

Additional Advantages of Using VLANs

In addition to reducing traffic and thereby speeding up the network, VLANs have the following advantages:

- VLANs provide segmentation services traditionally provided by routers in LAN configurations, thereby reducing hardware equipment costs.
- Packets coupled to a VLAN can be reliably identified and sorted into different domains. You can contain broadcasts within parts of the network, thereby freeing up network resources. For example, when a DHCP server is plugged into a switch and starts broadcasting its presence, you can prevent some hosts from accessing it by using VLANs to split up the network.
- For security issues, VLANs provide granular control of the network because each VLAN is identified by a single IP subnetwork. All packets passing in and out of a VLAN are consistently tagged with the VLAN ID of that VLAN, thereby providing easy identification, because a VLAN ID on a packet cannot be altered. (For a switch that runs Junos OS that does not support ELS, we recommend that you avoid using 1 as a VLAN ID, because that ID is a default value.)
- VLANs react quickly to host relocation—this is also due to the persistent VLAN tag on packets.
- On an Ethernet LAN, all network nodes must be physically connected to the same network. In VLANs, the physical location of nodes is not important—you can group network devices in any way that makes sense for your organization, such as by department or business function, types of network nodes, or physical location.

Maximum VLANs and VLAN Members Per Switch

The number of VLANs supported per switch varies for each switch. Use the configuration-mode command **set vlans *vlan-name* *vlan-id* ?** to determine the maximum number of VLANs allowed on a switch. You cannot exceed this VLAN limit because you have to assign a specific ID number when you create a VLAN—you could overwrite one of the numbers, but you cannot exceed the limit.

You can, however, exceed the recommended VLAN member maximum for a switch.

On a switch that runs Junos OS that does not support the ELS configuration style, the maximum number of VLAN members allowed on the switch is eight times the maximum number of VLANs that the switch supports (vmember limit = $\text{vlan max} * 8$). If the configuration of the switch exceeds the recommended VLAN member maximum, a warning message appears when you commit the configuration. If you commit the configuration despite the warning, the commit succeeds, but there is a risk of the Ethernet switching process (eswd) failing as a result of memory allocation failure.

On a switch that runs Junos OS that supports ELS, the maximum number of VLAN members allowed on the switch is 24 times the maximum number of VLANs that the switch supports (vmember limit = $\text{vlan max} * 24$). If the configuration of the switch exceeds the recommended VLAN member maximum, a warning message appears in the system log (syslog).

A QFabric system supports up to 131,008 VLAN members (vmembers) on a single network node group, server node group, or redundant server node group. The number of vmembers is calculated by multiplying the maximum number of VLANs by 32.

For example, to calculate how many interfaces are required to support 4,000 VLANs, divide the maximum number of vmembers (128,000) by the number of configured VLANs (4,000). In this case, 32 interfaces are required.

On network Node groups and server Node groups, you can configure link aggregation groups (LAGs) across multiple interfaces. Each LAG and VLAN combination is considered a vmember.

A Virtual Chassis Fabric supports up to 512,000 vmembers. The number of vmembers is based on the number of VLANs, and the number of interfaces configured in each VLAN.

A Default VLAN Is Configured on Most Switches

Some switches that run Junos OS that do not support the ELS configuration style are preconfigured with a VLAN named **default** that does not tag packets and operates only with untagged packets. On these switches, each interface already belongs to the VLAN named **default** and all traffic uses this VLAN until you configure more VLANs and assign traffic to those VLANs.



NOTE: When a Juniper Networks QFX3500 or QFX3600 switch is interconnected with other switches in a Virtual Chassis configuration, each individual switch that is included as a member of the configuration is identified with a member ID. The member ID functions as an FPC slot number. When you are configuring interfaces for a Virtual Chassis configuration, you specify the appropriate member ID (0 through 9) as the slot element of the interface name. The default factory settings for a Virtual Chassis configuration include FPC 0 as a member of the default VLAN because FPC 0 is configured as part of the ethernet-switching family. In order to include FPC 1 through FPC 9 in the default VLAN, add the ethernet-switching family to the configurations for those interfaces.

Assigning Traffic to VLANs

You can assign traffic on any switch to a particular VLAN by referencing either the interface port of the traffic or the MAC addresses of devices sending traffic.



NOTE: Two logical interfaces that are configured on the same physical interface cannot be mapped to the same VLAN.

Assign VLAN Traffic According to the Interface Port Source

This method is most commonly used to assign traffic to VLANs. In this case, you specify that all traffic received on a particular switch interface is assigned to a specific VLAN. You configure this VLAN assignment when you configure the switch, by using either the VLAN number (called a VLAN ID) or by using the VLAN name, which the switch then translates into a numeric VLAN ID. This method is referred to simply as creating a VLAN because it is the most commonly used method.

Assign VLAN Traffic According to the Source MAC Address

In this case, all traffic received from a specific MAC address is forwarded to a specific egress interface (next hop) on the switch. MAC-based VLANs are either static (named MAC addresses configured one at a time) or dynamic (configured using a RADIUS server).

To configure a static MAC-based VLAN on a switch that supports ELS, see *Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure)*. To configure a static MAC-based VLAN on a switch that does not support ELS, see *Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure)*.

Forwarding VLAN Traffic

To pass traffic within a VLAN, the switch uses Layer 2 forwarding protocols, including IEEE 802.1Q spanning-tree protocols.

To pass traffic between two VLANs, the switch uses standard Layer 3 routing protocols, such as static routing, OSPF, and RIP. The same interfaces that support Layer 2 bridging protocols also support Layer 3 routing protocols, providing multilayer switching.

To pass traffic from a single device on an access port to a switch and then pass those packets on a trunk port, use the native mode configuration previously discussed under [“Trunk Mode” on page 38](#).

VLANs Communicate with Integrated Routing and Bridging Interfaces or Routed VLAN Interfaces

Traditionally, switches sent traffic to hosts that were part of the same broadcast domain (VLAN) but routers were needed to route traffic from one broadcast domain to another. Also, only routers performed other Layer 3 functions such as traffic engineering.

Switches that run Junos OS that supports the ELS configuration style perform inter-VLAN routing functions using an integrated routing and bridging (IRB) interface named `irb`, while switches that run Junos OS that does not support ELS perform these functions using a routed VLAN interface (RVI) named `vlan`. These interfaces detect both MAC addresses and IP addresses and route data to Layer 3 interfaces, thereby frequently eliminating the need to have both a switch and a router.



NOTE:

Related Documentation

- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Example: Setting Up Bridging with Multiple VLANs](#)
- [Understanding FCoE](#)
- [Interfaces Overview](#)

Example: Setting Up Basic Bridging and a VLAN on the QFX Series

The QFX Series products use bridging and virtual LANs (VLANs) to connect network devices—storage devices, file servers, and other LAN components—in a LAN and to segment the LAN into smaller bridging domains.

To segment traffic on a LAN into separate broadcast domains, you create separate virtual LANs (VLANs) on a switch. Each VLAN is a collection of network nodes. When you use VLANs, frames whose origin and destination are in the same VLAN are forwarded only within the local VLAN, and only frames not destined for the local VLAN are forwarded to other broadcast domains. VLANs thus limit the amount of traffic flowing across the entire LAN, reducing the possible number of collisions and packet retransmissions within the LAN.



NOTE: You cannot configure more than one logical interface that belongs to the same physical interface in the same bridge domain.

This example describes how to configure basic bridging and VLANs for the QFX Series:

- [Requirements on page 43](#)
- [Overview and Topology on page 43](#)
- [Configuration on page 44](#)
- [Verification on page 53](#)

Requirements

This example uses the following software and hardware components:

- Junos OS Release 11.1 or later for the QFX Series
- A configured and provisioned QFX Series product

Overview and Topology

To use a switch to connect network devices on a LAN, you must at a minimum configure bridging and VLANs. By default, bridging is enabled on all switch interfaces, all interfaces are in access mode, and all interfaces belong to a VLAN called **employee-vlan**, which is automatically configured. When you plug in access devices—such as desktop computers, file servers, and printers—they are joined immediately into the **employee-vlan** VLAN, and the LAN is up and running.

The topology used in this example consists of a single QFX3500 switch, with a total of 48 10-Gbps Ethernet ports. (For the purposes of this example, the QSFP+ ports Q0-Q3, which are ports xe-0/1/0 through xe-0/1/15, are excluded.) You use the ports to connect devices that have their own power sources. Table 1 details the topology used in this configuration example.

Table 4: Components of the Basic Bridging Configuration Topology

Property	Settings
Switch hardware	QFX3500 switch, with 48 10-Gbps Ethernet ports
VLAN name	employee-vlan
VLAN ID	10
Connections to file servers	xe-0/0/17 and xe-0/0/18
Direct connections to desktop PCs and laptops	xe-0/0/0 through xe-0/0/16
Connections to integrated printer/fax/copier machines	xe-0/0/19 through xe-0/0/40
Unused ports	xe-0/0/41 through xe-0/0/47

Configuration

CLI Quick Configuration To quickly configure a VLAN, copy the following commands and paste them into the switch terminal window:

```
[edit]
set vlans employee-vlan vlan-id 10
set interfaces xe-0/0/0 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/1 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/2 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/3 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/4 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/5 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/6 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/7 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/8 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/9 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/10 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/11 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/12 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/13 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/14 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/15 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/16 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/17 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/18 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/19 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/22 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/23 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/24 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/25 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/26 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/27 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/28 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/29 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/30 unit 0 family ethernet-switching vlan members employee-vlan
```

```

set interfaces xe-0/0/31 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/32 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/33 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/34 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/35 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/36 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/37 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/38 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/39 unit 0 family ethernet-switching vlan members employee-vlan
set interfaces xe-0/0/40 unit 0 family ethernet-switching vlan members employee-vlan

```

Step-by-Step Procedure

To set up basic bridging and a VLAN:

1. Create a VLAN named employee-vlan and specify the VLAN ID of 10 for it:

```

[edit vlans]
user@switch# set employee-vlan vlan-id 10

```

2. Assign interfaces xe-0/0/0 through xe-0/0/40 to the employee-vlan VLAN:

```

[edit interface]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/1 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/2 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/3 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/4 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/5 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/6 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/7 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/8 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/9 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/10 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/11 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/12 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/13 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/14 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/15 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/16 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/17 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/18 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/19 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/20 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/21 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/22 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/23 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/24 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/25 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/26 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/27 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/28 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/29 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/30 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/31 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/32 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/33 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/34 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/35 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/36 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/37 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/38 unit 0 family ethernet-switching vlan members employee-vlan
user@switch# set xe-0/0/39 unit 0 family ethernet-switching vlan members employee-vlan

```

```
user@switch# set xe-0/0/40 unit 0 family ethernet-switching vlan members employee-vlan
```

3. Connect the two file servers to ports xe-0/0/17 and xe-0/0/18.
4. Connect the desktop PCs and laptops to ports xe-0/0/0 through xe-0/0/16.
5. Connect the integrated printer/fax/copier machines to ports xe-0/0/19 through xe-0/0/40.

Results Check the results of the configuration:

```
user@switch> show configuration
xe-0/0/0 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/1 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/2 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/3 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/4 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/5 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/6 {
  unit 0 {
    family ethernet-switching {
```

```
        vlan {
            members employee-vlan;
        }
    }
}
xe-0/0/7 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/8 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/9 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/10 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/11 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/12 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
```

```
xe-0/0/13 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/14 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/15 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/16 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/17 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/18 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
xe-0/0/19 {
  unit 0 {
    family ethernet-switching {
      vlan {
        members employee-vlan;
      }
    }
  }
}
```

```
    }  
  }  
  xe-0/0/20 {  
    unit 0 {  
      family ethernet-switching {  
        vlan {  
          members employee-vlan;  
        }  
      }  
    }  
  }  
  xe-0/0/21 {  
    unit 0 {  
      family ethernet-switching {  
        vlan {  
          members employee-vlan;  
        }  
      }  
    }  
  }  
  xe-0/0/22 {  
    unit 0 {  
      family ethernet-switching {  
        vlan {  
          members employee-vlan;  
        }  
      }  
    }  
  }  
  xe-0/0/23 {  
    unit 0 {  
      family ethernet-switching {  
        vlan {  
          members employee-vlan;  
        }  
      }  
    }  
  }  
  xe-0/0/24 {  
    unit 0 {  
      family ethernet-switching {  
        vlan {  
          members employee-vlan;  
        }  
      }  
    }  
  }  
  xe-0/0/25 {  
    unit 0 {  
      family ethernet-switching {  
        vlan {  
          members employee-vlan;  
        }  
      }  
    }  
  }  
  xe-0/0/26 {  
    unit 0 {  
      family ethernet-switching {  
        vlan {  
          members employee-vlan;  
        }  
      }  
    }  
  }
```



```
    }  
  }  
}  
xe-0/0/27 {  
  unit 0 {  
    family ethernet-switching {  
      vlan {  
        members employee-vlan;  
      }  
    }  
  }  
}  
xe-0/0/28 {  
  unit 0 {  
    family ethernet-switching {  
      vlan {  
        members employee-vlan;  
      }  
    }  
  }  
}  
xe-0/0/29 {  
  unit 0 {  
    family ethernet-switching {  
      vlan {  
        members employee-vlan;  
      }  
    }  
  }  
}  
xe-0/0/30 {  
  unit 0 {  
    family ethernet-switching {  
      vlan {  
        members employee-vlan;  
      }  
    }  
  }  
}  
xe-0/0/31 {  
  unit 0 {  
    family ethernet-switching {  
      vlan {  
        members employee-vlan;  
      }  
    }  
  }  
}  
xe-0/0/32 {  
  unit 0 {  
    family ethernet-switching {  
      vlan {  
        members employee-vlan;  
      }  
    }  
  }  
}  
xe-0/0/33 {  
  unit 0 {  
    family ethernet-switching {
```

```
        vlan {
            members employee-vlan;
        }
    }
}
xe-0/0/34 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/35 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/36 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/37 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/38 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/39 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members employee-vlan;
            }
        }
    }
}
xe-0/0/40 {
```

```
unit 0 {  
    family ethernet-switching {  
        vlan {  
            members employee-vlan;  
        }  
    }  
}
```

Verification

To verify that switching is operational and that **employee-vlan** has been created, perform these tasks:

- [Verifying That the VLAN Has Been Created on page 53](#)
- [Verifying That Interfaces Are Associated with the Proper VLANs on page 54](#)

Verifying That the VLAN Has Been Created

Purpose Verify that the VLAN named **employee-vlan** has been created on the switch.

Action List all VLANs configured on the switch:

```
user@switch> show vlans
Routing instance      VLAN name      Tag      Interfaces
default-switch        employee-vlan   10
                      xe-0/0/0.0
                      xe-0/0/1.0
                      xe-0/0/2.0
                      xe-0/0/3.0
                      xe-0/0/4.0
                      xe-0/0/5.0
                      xe-0/0/6.0
                      xe-0/0/7.0
                      xe-0/0/8.0
                      xe-0/0/9.0
                      xe-0/0/10.0
                      xe-0/0/11.0
                      xe-0/0/12.0
                      xe-0/0/13.0
                      xe-0/0/14.0
                      xe-0/0/15.0
                      xe-0/0/16.0
                      xe-0/0/17.0
                      xe-0/0/18.0
                      xe-0/0/19.0
                      xe-0/0/20.0
                      xe-0/0/21.0
                      xe-0/0/22.0
                      xe-0/0/23.0
                      xe-0/0/24.0
                      xe-0/0/25.0
                      xe-0/0/26.0
                      xe-0/0/27.0
                      xe-0/0/28.0
                      xe-0/0/29.0
                      xe-0/0/30.0
                      xe-0/0/31.0
                      xe-0/0/32.0
                      xe-0/0/33.0
                      xe-0/0/34.0
                      xe-0/0/35.0
                      xe-0/0/36.0
                      xe-0/0/37.0
                      xe-0/0/38.0
                      xe-0/0/39.0
                      xe-0/0/40.0
...

```

Meaning The `show vlans` command lists the VLANs configured on the switch. This output shows that the VLAN `employee-vlan` has been created.

Verifying That Interfaces Are Associated with the Proper VLANs

Purpose Verify that Ethernet switching is enabled on switch interfaces and that all interfaces are included in the VLAN.

Action List all interfaces on which switching is enabled:

```

user@switch> show ethernet-switching interfaces
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/0.0                65535                untagged
                        employee-vlan 10
                        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/1.0                65535                untagged
                        employee-vlan 10
                        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/2.0                65535                untagged
                        employee-vlan 10
                        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/3.0                65535                untagged
                        employee-vlan 10
                        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/4.0                65535                untagged
                        employee-vlan 10
                        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/5.0                65535                untagged
                        employee-vlan 10
                        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/6.0                65535                untagged
                        employee-vlan 10
                        65535 Discarding
Routing Instance Name : default-switch

```

```

Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/7.0
      employee-vlan 10
                        65535
                        Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/8.0
      employee-vlan 10
                        65535
                        Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/9.0
      employee-vlan 10
                        65535
                        Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/10.0
      employee-vlan 10
                        65535
                        Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/11.0
      employee-vlan 10
                        65535
                        Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/12.0
      employee-vlan 10
                        65535
                        Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/13.0
      employee-vlan 10
                        65535
                        Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members   limit state   interface flags
xe-0/0/14.0
      employee-vlan 10
                        65535
                        Discarding

```

```

        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members          limit  state      interface flags
xe-0/0/15.0              65535              untagged
        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members          limit  state      interface flags
xe-0/0/16.0              65535              untagged
        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members          limit  state      interface flags
xe-0/0/17.0              65535              untagged
        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members          limit  state      interface flags
xe-0/0/18.0              65535              untagged
        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members          limit  state      interface flags
xe-0/0/19.0              65535              untagged
        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members          limit  state      interface flags
xe-0/0/20.0              65535              untagged
        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members          limit  state      interface flags
xe-0/0/21.0              65535              untagged
        employee-vlan 10
        65535 Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )

```

```

Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/22.0
    employee-vlan 10
                                65535   Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/23.0
    employee-vlan 10
                                65535   Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/24.0
    employee-vlan 10
                                65535   Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/25.0
    employee-vlan 10
                                65535   Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/26.0
    employee-vlan 10
                                65535   Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/27.0
    employee-vlan 10
                                65535   Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/28.0
    employee-vlan 10
                                65535   Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members          limit state interface flags
xe-0/0/29.0
    employee-vlan 10
                                65535   Discarding

```



```

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags
xe-0/0/30.0   employee-vlan 10   65535   Discarding

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags
xe-0/0/31.0   employee-vlan 10   65535   Discarding

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags
xe-0/0/32.0   employee-vlan 10   65535   Discarding

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags
xe-0/0/33.0   employee-vlan 10   65535   Discarding

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags
xe-0/0/34.0   employee-vlan 10   65535   Discarding

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags
xe-0/0/35.0   employee-vlan 10   65535   Discarding

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags
xe-0/0/36.0   employee-vlan 10   65535   Discarding

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical      Vlan      TAG  MAC      STP      Logical      Tagging
interface    members   limit state   interface flags

```

```

xe-0/0/37.0          65535          untagged
                    employee-vlan 10
                    65535          Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members limit state interface flags
xe-0/0/38.0          65535          untagged
                    employee-vlan 10
                    65535          Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members limit state interface flags
xe-0/0/39.0          65535          untagged
                    employee-vlan 10
                    65535          Discarding
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down )
Logical   Vlan   TAG   MAC   STP   Logical   Tagging
interface members limit state interface flags
xe-0/0/40.0          65535          untagged
                    employee-vlan 10
                    65535          Discarding
...

```

Meaning The `show ethernet-switching interfaces` command lists all interfaces on which switching is enabled (in the **Logical interface** column), along with the VLANs that are active on the interfaces (in the **VLAN members** column). The output in this example shows all the connected interfaces, xe-0/0/0 through xe-0/0/40, are all part of VLAN **employee-vlan**. Notice that the interfaces listed are the logical interfaces, not the physical interfaces. For example, the output shows xe-0/0/0.0 instead of xe-0/0/0. This is because Junos OS creates VLANs on logical interfaces, not directly on physical interfaces.

Related Documentation

- [Example: Setting Up Bridging with Multiple VLANs](#)
- [Understanding Bridging and VLANs on page 35](#)

Example: Setting Up Bridging with Multiple VLANs

The QFX Series products use bridging and virtual LANs (VLANs) to connect network devices in a LAN—storage devices, file servers, and other network components—and to segment the LAN into smaller bridging domains.

To segment traffic on a LAN into separate broadcast domains, you create separate virtual LANs (VLANs) on a switch. Each VLAN is a collection of network nodes. When you use VLANs, frames whose origin and destination are in the same VLAN are forwarded only within the local VLAN, and only frames not destined for the local VLAN are forwarded to other broadcast domains. VLANs thus limit the amount of traffic flowing across the entire LAN, reducing the possible number of collisions and packet retransmissions within the LAN.

This example describes how to configure bridging for the QFX Series and how to create two VLANs to segment the LAN:



NOTE: This task supports the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*. If your switch runs software that does not support ELS, see *Example: Setting Up Bridging with Multiple VLANs*.

- [Requirements on page 61](#)
- [Overview and Topology on page 61](#)
- [Configuration on page 62](#)
- [Verification on page 64](#)

Requirements

This example uses the following hardware and software components:

- A configured and provisioned QFX3500 switch
- Junos OS Release 13.2X50-D15 or later for the QFX Series

Overview and Topology

Switches connect all devices in an office or data center into a single LAN to provide sharing of common resources such as file servers. The default configuration creates a single VLAN, and all traffic on the switch is part of that broadcast domain. Creating separate network segments reduces the span of the broadcast domain and enables you to group related users and network resources without being limited by physical cabling or by the location of a network device in the building or on the LAN.

This example shows a simple configuration to illustrate the basic steps for creating two VLANs on a single switch. One VLAN, called **sales**, is for the sales and marketing group, and a second, called **support**, is for the customer support team. The sales and support groups each have their own dedicated file servers and other resources. For the switch ports to be segmented across the two VLANs, each VLAN must have its own broadcast domain, identified by a unique name and tag (VLAN ID). In addition, each VLAN must be on its own distinct IP subnet.

The topology used in this example consists of a single QFX3500 switch, with a total of 48 10-Gbps Ethernet ports. (For the purposes of this example, the QSFP+ ports Q0-Q3, which are ports xe-0/1/0 through xe-0/1/15, are excluded.)

Table 5: Components of the Multiple VLAN Topology

Property	Settings
Switch hardware	QFX3500 switch configured with 48 10-Gbps Ethernet ports (xe-0/0/0 through xe-0/0/47)

Table 5: Components of the Multiple VLAN Topology (*continued*)

Property	Settings
VLAN names and tag IDs	sales , tag 100 support , tag 200
VLAN subnets	sales : 192.0.2.0/25 (addresses 192.0.2.1 through 192.0.2.126) support : 192.0.2.128/25 (addresses 192.0.2.129 through 192.0.2.254)
Interfaces in VLAN sales	File servers: xe-0/0/20 and xe-0/0/21
Interfaces in VLAN support	File servers: xe-0/0/46 and xe-0/0/47
Unused interfaces	xe-0/0/2 and xe-0/0/25

This configuration example creates two IP subnets, one for the sales VLAN and the second for the support VLAN. The switch bridges traffic within a VLAN. For traffic passing between two VLANs, the switch routes the traffic using a Layer 3 routing interface on which you have configured the address of the IP subnet.

To keep the example simple, the configuration steps show only a few devices in each of the VLANs. Use the same configuration procedure to add more LAN devices.

Configuration

CLI Quick Configuration

To quickly configure Layer 2 switching for the two VLANs (**sales** and **support**) and to quickly configure Layer 3 routing of traffic between the two VLANs, copy the following commands and paste them into the switch terminal window:

```
[edit]
set interfaces xe-0/0/0 unit 0 family ethernet-switching vlan members sales
set interfaces xe-0/0/3 unit 0 family ethernet-switching vlan members sales
set interfaces xe-0/0/22 unit 0 family ethernet-switching vlan members sales
set interfaces xe-0/0/20 unit 0 description "Sales file server port"
set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members sales
set interfaces xe-0/0/24 unit 0 family ethernet-switching vlan members support
set interfaces xe-0/0/26 unit 0 family ethernet-switching vlan members support
set interfaces xe-0/0/44 unit 0 family ethernet-switching vlan members support
set interfaces xe-0/0/46 unit 0 description "Support file server port"
set interfaces xe-0/0/46 unit 0 family ethernet-switching vlan members support
set interfaces vlan unit 0 family inet address 192.0.2.0/25
set interfaces vlan unit 1 family inet address 192.0.2.128/25
set vlans sales l3-interface irb.0
set vlans sales vlan-id 100
set vlans support vlan-id 200
set vlans support l3-interface irb.1
```

Step-by-Step Procedure Configure the switch interfaces and the VLANs to which they belong. By default, all interfaces are in access mode, so you do not have to configure the port mode.

1. Configure the interface for the file server in the **sales** VLAN:

```
[edit interfaces xe-0/0/20 unit 0]
user@switch# set description "Sales file server port"
user@switch# set family ethernet-switching vlan members sales
```
2. Configure the interface for the file server in the **support** VLAN:

```
[edit interfaces xe-0/0/46 unit 0]
user@switch# set description "Support file server port"
user@switch# set family ethernet-switching vlan members support
```
3. Create the subnet for the **sales** broadcast domain:

```
[edit interfaces]
user@switch# set vlan unit 0 family inet address 192.0.2.1/25
```
4. Create the subnet for the **support** broadcast domain:

```
[edit interfaces]
user@switch# set vlan unit 1 family inet address 192.0.2.129/25
```
5. Configure the VLAN tag IDs for the **sales** and **support** VLANs:

```
[edit vlans]
user@switch# set sales vlan-id 100
user@switch# set support vlan-id 200
```
6. To route traffic between the **sales** and **support** VLANs, define the interfaces that are members of each VLAN and associate a Layer 3 interface:

```
[edit vlans]
user@switch# set sales l3-interface irb.0
user@switch# set support l3-interface irb.1
```

Display the results of the configuration:

```
user@switch> show configuration
interfaces {
  xe-0/0/20 {
    unit 0 {
      description "Sales file server port";
      family ethernet-switching {
        vlan members sales;
      }
    }
  }
  xe-0/0/46 {
    unit 0 {
      description "Support file server port";
      family ethernet-switching {
        vlan members support;
      }
    }
  }
  vlans {
    unit 0 {
      family inet address 192.0.2.1/25;
    }
    unit 1 {
      family inet address 192.0.2.129/25;
    }
  }
}
```

```

    }
  }
}
vllans {
  sales {
    vlan-id 100;
    interface xe-0/0/0.0;
    interface xe-0/0/3.0;
    interface xe-0/0/20.0;
    interface xe-0/0/22.0;
    l3-interface irb0;
  }
  support {
    vlan-id 200;
    interface xe-0/0/24.0;
    interface xe-0/0/26.0;
    interface xe-0/0/44.0;
    interface xe-0/0/46.0;
    l3-interface irb1;
  }
}
}

```



TIP: To quickly configure the **sales** and **support** VLAN interfaces, issue the **load merge terminal** command. Then copy the hierarchy and paste it into the switch terminal window.

Verification

Verify that the **sales** and **support** VLANs have been created and are operating properly, perform these tasks:

- [Verifying That the VLANs Have Been Created and Associated with the Correct Interfaces on page 64](#)
- [Verifying That Traffic Is Being Routed Between the Two VLANs on page 65](#)
- [Verifying That Traffic Is Being Switched Between the Two VLANs on page 65](#)

Verifying That the VLANs Have Been Created and Associated with the Correct Interfaces

Purpose Verify that the **sales** and **support** VLANs have been created on the switch and that all connected interfaces on the switch are members of the correct VLAN.

Action To list all VLANs configured on the switch, use the **show vlans** command:

```

user@switch> show vlans
Name      Tag      Interfaces
default
          xe-0/0/1.0, xe-0/0/2.0, xe-0/0/4.0, xe-0/0/5.0,
          xe-0/0/6.0, xe-0/0/7.0, xe-0/0/8.0, xe-0/0/9.0,
          xe-0/0/10.0*, xe-0/0/11.0, xe-0/0/12.0, xe-0/0/13.0*,
          xe-0/0/14.0, xe-0/0/15.0, xe-0/0/16.0, xe-0/0/17.0,

```

```

xe-0/0/18.0, xe-0/0/19.0, xe-0/0/21.0, xe-0/0/23.0*,
xe-0/0/25.0, xe-0/0/27.0, xe-0/0/28.0, xe-0/0/29.0,
xe-0/0/30.0, xe-0/0/31.0, xe-0/0/32.0, xe-0/0/33.0,
xe-0/0/34.0, xe-0/0/35.0, xe-0/0/36.0, xe-0/0/37.0,
xe-0/0/38.0, xe-0/0/39.0, xe-0/0/40.0, xe-0/0/41.0,
xe-0/0/42.0, xe-0/0/43.0, xe-0/0/45.0, xe-0/0/47.0,
xe-0/1/0.0*, xe-0/1/1.0*, xe-0/1/2.0*, xe-0/1/3.0*

sales      100
           xe-0/0/0.0*, xe-0/0/3.0, xe-0/0/20.0, xe-0/0/22.0

support    200
           xe-0/0/0.24, xe-0/0/26.0, xe-0/0/44.0, xe-0/0/46.0*

mgmt
           me0.0*

```

Meaning The **show vlans** command lists all VLANs configured on the switch and which interfaces are members of each VLAN. This command output shows that the **sales** and **support** VLANs have been created. The **sales** VLAN has a tag ID of 100 and is associated with interfaces **xe-0/0/0.0**, **xe-0/0/3.0**, **xe-0/0/20.0**, and **xe-0/0/22.0**. VLAN **support** has a tag ID of 200 and is associated with interfaces **xe-0/0/24.0**, **xe-0/0/26.0**, **xe-0/0/44.0**, and **xe-0/0/46.0**.

Verifying That Traffic Is Being Routed Between the Two VLANs

Purpose Verify routing between the two VLANs.

Action List the Layer 3 routes in the switch Address Resolution Protocol (ARP) table:

```

user@switch> show arp
MAC Address      Address      Name      Flags
00:00:0c:06:2c:0d 192.0.2.3    vlan.0    None
00:13:e2:50:62:e0 192.0.2.11   vlan.1    None

```

Meaning Sending IP packets on a multiaccess network requires mapping from an IP address to a MAC address (the physical or hardware address). The ARP table displays the mapping between the IP address and MAC address for both **vlan.0** (associated with **sales**) and **vlan.1** (associated with **support**). These VLANs can route traffic to each other.

Verifying That Traffic Is Being Switched Between the Two VLANs

Purpose Verify that learned entries are being added to the Ethernet switching table.

Action List the contents of the Ethernet switching table:

```

user@switch> show ethernet-switching table

Ethernet-switching table: 8 entries, 5 learned
VLAN      MAC address      Type      Age Interfaces
default   *                Flood     - All-members
default   00:00:05:00:00:01 Learn     - xe-0/0/10.0
default   00:00:5e:00:01:09 Learn     - xe-0/0/13.0

```

default	00:19:e2:50:63:e0	Learn	- xe-0/0/23.0
sales	*	Flood	- All-members
sales	00:00:5e:00:07:09	Learn	- xe-0/0/0.0
support	*	Flood	- All-members
support	00:00:5e:00:01:01	Learn	- xe-0/0/46.0

Meaning The output shows that learned entries for the **sales** and **support** VLANs have been added to the Ethernet switching table, and are associated with interfaces **xe-0/0/0.0** and **xe-0/0/46.0**. Even though the VLANs were associated with more than one interface in the configuration, these interfaces are the only ones that are currently operating.

Related Documentation

- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Understanding Bridging and VLANs on page 35](#)

Configuring VLANs

Switches use VLANs to make logical groupings of network nodes with their own broadcast domains. You can use VLANs to limit the traffic flowing across the entire LAN and reduce collisions and packet retransmissions.



NOTE: This task supports the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*. If your switch runs software that does not support ELS, see *Configuring VLANs*.



NOTE: Two logical interfaces that are configured on the same physical interface cannot be mapped to the same VLAN.

For each endpoint on the VLAN, configure the following VLAN parameters on the corresponding interface:

1. Specify the description of the VLAN:

```
[edit interfaces interface-name unit 0]
user@switch# set description vlan-description
```

2. Specify the unique name of the VLAN:



NOTE: Switches that run Junos OS with the ELS configuration style do not support a default VLAN. Therefore, on such switches, you must explicitly configure at least one VLAN, even if your network is simple and you want only one broadcast domain to exist.

```
[edit interfaces interface-name unit 0]
user@switch# set family ethernet-switching vlan members vlan-name
```

3. Create the subnet for the VLAN:

```
[edit interfaces]
```



```

user@switch# set vlan unit 0 family inet address ip-address
4. Configure the VLAN tag ID or VLAN ID list for the VLAN:

[edit vlans]
user@switch# set vlan-name vlan-id vlan-id-number
or
[edit vlans]
user@switch# set vlan-name vlan-id-list [vlan-ids | vlan-id--vlan-id-]

```

Related Documentation

- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Configuring IRB Interfaces on page 72](#)
- [Creating a Series of Tagged VLANs](#)
- [Understanding Bridging and VLANs on page 35](#)

Configuring the Native VLAN Identifier (CLI Procedure)



NOTE: This task uses Junos OS for EX Series switches and Junos OS for QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Configuring the Native VLAN Identifier (CLI Procedure)*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

Switches can receive and forward routed or bridged Ethernet frames with 802.1Q VLAN tags. Typically, trunk ports, which connect switches to each other, accept untagged control packets but do not accept untagged data packets. You can enable a trunk port to accept untagged data packets by configuring a native VLAN ID on the interface on which you want the untagged data packets to be received. The logical interface on which untagged packets are to be received must be configured with the same VLAN ID as the native VLAN ID configured on the physical interface.

To configure the native VLAN ID by using the command-line interface (CLI):

1. On the interface on which you want untagged data packets to be received, set the interface mode to **trunk**, which specifies that the interface is in multiple VLANs and can multiplex traffic between different VLANs.:

```

[edit interfaces]
user@switch# set interface-name unit logical-unit-number family
ethernet-switching interface-mode trunk

```

2. Configure the native VLAN ID:

```

[edit interfaces]
user@switch# set interface-name native-vlan-id vlan-id

```

3. Specify that the logical interface that will receive the untagged data packets is a member of the native VLAN:

```

[edit interfaces]
user@switch# set interface-name unit logical-unit-number family
ethernet-switching vlan members vlan-id

```

**Related
Documentation**

- *Understanding Bridging and VLANs on EX Series Switches*
- *Example: Connecting Access Switches to a Distribution Switch*
- *Example: Setting Up Basic Bridging and a VLAN for an EX Series Switch*
- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)

Creating a Series of Tagged VLANs

When you divide an Ethernet LAN into multiple VLANs, each VLAN is assigned a unique IEEE 802.1Q tag. This tag is associated with each frame in the VLAN, and the network nodes receiving the traffic can use the tag to identify which VLAN a frame is associated with.

Instead of configuring VLANs and 802.1Q tags one at a time for a trunk interface, you can configure a VLAN range to create a series of tagged VLANs.

When an Ethernet LAN is divided into VLANs, each VLAN is identified by a unique 802.1Q tag. The tag is applied to all frames so that the network nodes receiving the frames can detect which VLAN the frames belong to. Trunk ports, which multiplex traffic among a number of VLANs, use the tag to determine the origin of frames and where to forward them.

For example, you could configure the VLAN **employee** and specify a tag range of **10 through 12**. This creates the following VLANs and tags:

- VLAN **employee-10**, tag 10
- VLAN **employee-11**, tag 11
- VLAN **employee-12**, tag 12

Creating tagged VLANs in a series has the following limitations:

- Layer 3 interfaces do not support this feature.
- Because an access interface can only support one VLAN member, access interfaces also do not support this feature.



NOTE: This task uses Junos OS for Junos OS for QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Creating a Series of Tagged VLANs*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

To configure a series of tagged VLANs using the CLI (here, the VLAN is **employee**):

1. Configure the series (here, a VLAN series from 120 through 130):

```
[edit]
user@switch# set vlans employee vlan-id-list [ 120-130 ]
```

2. Associate a series of tagged VLANs when you configure an interface in one of two ways:

- Include the name of the series:

```
[edit interfaces]
user@switch# set interfaces xe-0/0/22.0 family ethernet-switching vlanmembers employee
```

- Include the VLAN range:

```
[edit interfaces]
user@switch# set interfaces xe-0/0/22.0 family ethernet-switching vlan members 120-130
```

Associating a series of tagged VLANs to an interface by name or by VLAN range the same result: VLANs **__employee_120__** through **__employee_130__** are created.



NOTE: When a series of VLANs is created using the `vlan-id-list` command, the VLAN names are preceded and followed by a double underscore.

Related Documentation

- [Verifying That a Series of Tagged VLANs Has Been Created](#)
- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Example: Setting Up Bridging with Multiple VLANs on page 60](#)
- [Understanding Bridging and VLANs on page 35](#)

Understanding Integrated Routing and Bridging

To segment traffic on a LAN into separate broadcast domains, you create separate virtual LANs (VLANs). VLANs limit the amount of traffic flowing across the entire LAN, reducing the possible number of collisions and packet retransmissions within the LAN. For example, you might want to create a VLAN that includes the employees in a department and the resources that they use often, such as printers, servers, and so on.

Of course, you also want to allow these employees to communicate with people and resources in other VLANs. To forward packets between VLANs, you normally you need a router that connects the VLANs. However, you can accomplish this forwarding on a switch without using a router by configuring an integrated routing and bridging (IRB) interface. (These interfaces are also called routed VLAN interfaces, or RVIs). Using this approach reduces complexity and avoids the costs associated with purchasing, installing, managing, powering, and cooling another device.

An IRB is a special type of Layer 3 virtual interface named **vlan**. Like normal Layer 3 interfaces, the **vlan** interface needs a logical unit number with an IP address. In fact, to be useful an IRB needs at least two logical units and two IP addresses—you must create units with addresses in each of the subnets associated with the VLANs between which

you want traffic to be routed. That is, if you have two VLANs (for example, VLAN **red** and VLAN **blue**) with corresponding subnets, your IRB must have a logical unit with an address in the subnet for **red** and a logical unit with an address in the subnet for **blue**. The switch automatically creates direct routes to these subnets and uses these routes to forward traffic between VLANs.

[Table 6 on page 71](#) shows values you might use when configuring an IRB:

Table 6: Sample IRB Values

Property	Settings
VLAN names and tags (IDs)	blue , ID 100 red , ID 200
Subnets associated with VLANs	blue : 192.0.2.0/25 (addresses 192.0.2.1 through 192.0.2.126) red : 192.0.2.128/25 (addresses 192.0.2.129 through 192.0.2.254)
IRB name	interface irb
IRB units and addresses	logical unit 100: 192.0.2.1/25 logical unit 200: 192.0.2.129/25

For the sake of consistency and to avoid confusion, [Table 6 on page 71](#) shows IRB logical unit numbers that match the IDs of the corresponding VLANs. However, you do not have to assign logical unit numbers that match the VLAN IDs—you can use any values for the units. To bind the logical units of the IRB to the appropriate VLANs, you use the **l3-interface** statement.

Because IRBs operate at Layer 3, you can use Layer 3 services such as firewall filters or CoS rewriting with them.

[Table 7 on page 71](#) shows the number of IRBs/RVIs that each QFX platform supports.

Table 7: Number of Supported IRBs/RVIs by Platform

Platform	Number of Supported IRBs/RVIs
QFX5100	4000
EX4600	4000
OCX1100	4000
QFX3500	1200
QFX3000-G	1024
QFX3000-M	1024

Related Documentation • [Example: Configuring Routing Between VLANs on One Switch on page 73](#)

Configuring IRB Interfaces

Integrated routing and bridging (IRB) interfaces enable a switch to recognize which packets are being sent to local addresses so that they are bridged whenever possible and are routed only when needed. Whenever packets can be switched instead of routed, several layers of processing are eliminated. Switching also reduces the number of address look-ups.



NOTE: In versions of Junos OS that do not support Enhanced Layer 2 Software (ELS), this type of interface is called a routed VLAN interface (RVI).

To configure the routed VLAN interface:

1. Create the VLAN by assigning it a name and a VLAN ID:

```
[edit]
user@switch# set vlans support vlan-id 111
```

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

```
[edit]
user@switch# set interfaces ge-0/0/18 unit 0 family ethernet-switching vlan members
support
```

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

```
[edit]
user@switch# set interfaces irb unit 111 family inet address 111.111.111.1/24
```

4. Bind a Layer 3 interface with the VLAN:

```
[edit]
user@switch# set vlans support l3-interface irb.111
```



NOTE: If you are using a version of Junos OS that does not support ELS, you create a Layer 3 virtual interface named **vlan**



NOTE: Layer 3 interfaces on trunk ports allow the interface to transfer traffic between multiple VLANs. Within a VLAN, traffic is bridged, while across VLANs, traffic is routed.

You can display the configuration settings:

```
user@switch> show interfaces irb terse
Interface      Admin Link Proto  Local          Remote
vlan           up    up
irb.111        up    up   inet   111.111.111.1/24

user@switch> show vlans
Name      Tag      Interfaces
default
None
```

```

employee-vlan 20
                ge-1/0/0.0, ge-1/0/1.0, ge-1/0/2.0
marketing      40
                ge-1/0/10.0, ge-1/0/20.0, ge-1/0/30.0
support        111
                ge-0/0/18.0
mgmt
                bme0.32769, bme0.32771*

user@switch> show ethernet-switching table
Ethernet-switching table: 1 entries, 0 learned
  VLAN      MAC address      Type      Age Interfaces
  support    00:19:e2:50:95:a0 Static      - Router

```

Related Documentation

- [Understanding Integrated Routing and Bridging on page 70](#)

Example: Configuring Routing Between VLANs on One Switch

To segment traffic on a LAN into separate broadcast domains, you create separate virtual LANs (VLANs). For example, you might want to create a VLAN that includes the employees in a department and the resources that they use often, such as printers, servers, and so on.

Of course, you also want to allow these employees to communicate with people and resources in other VLANs. To forward packets between VLANs you normally you need a router that connects the VLANs. However, you can accomplish this on a Juniper Networks switch without using a router by configuring an integrated routing and bridging (IRB) interface (also known as a routed VLAN interface—or RVI—in versions of Junos OS that do not support Enhanced Layer 2 Software). Using this approach reduces complexity and avoids the costs associated with purchasing, installing, managing, powering, and cooling another device.

- [Requirements on page 73](#)
- [Overview and Topology on page 73](#)
- [Configure Layer 2 switching for two VLANs on page 74](#)
- [Verification on page 77](#)

Requirements

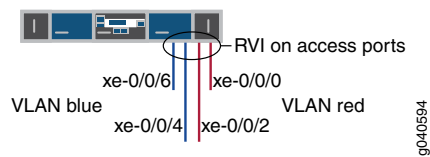
This example uses the following hardware and software components:

- One switch
- Junos OS Release 11.1 or later

Overview and Topology

This example uses an IRB to route traffic between two VLANs on the same switch. The topology is shown in [Figure 4 on page 74](#).

Figure 4: IRB with One Switch



This example shows a simple configuration to illustrate the basic steps for creating two VLANs on a single switch and configuring an IRB to enable routing between the VLANs. One VLAN, called **blue**, is for the sales and marketing group, and a second, called **red**, is for the customer support team. The sales and support groups each have their own file servers and wireless access points. Each VLAN must have a unique name, tag (VLAN ID), and distinct IP subnet. [Table 8 on page 74](#) lists the components of the sample topology.

Table 8: Components of the Multiple VLAN Topology

Property	Settings
VLAN names and tag IDs	blue , ID 100 red , ID 200
Subnets associated with VLANs	blue : 192.0.2.0/25 (addresses 192.0.2.1 through 192.0.2.126) red : 192.0.2.128/25 (addresses 192.0.2.129 through 192.0.2.254)
Interfaces in VLAN blue	Sales server port: xe-0/0/4 Sales wireless access points: xe-0/0/6
Interfaces in VLAN red	Support server port: xe-0/0/0 Support wireless access points: xe-0/0/2
IRB name	interface irb
IRB units and addresses	logical unit 100: 192.0.2.1/25 logical unit 200: 192.0.2.129/25

This configuration example creates two IP subnets, one for the blue VLAN and the second for the red VLAN. The switch bridges traffic within the VLANs. For traffic passing between two VLANs, the switch routes the traffic using an IRB on which you have configured addresses in each IP subnet.

To keep the example simple, the configuration steps show only a few interfaces and VLANs. Use the same configuration procedure to add more interfaces and VLANs. By default, all interfaces are in access mode, so you do not have to configure the port mode.

Configure Layer 2 switching for two VLANs

CLI Quick Configuration

To quickly configure Layer 2 switching for the two VLANs (**blue** and **red**) and to quickly configure Layer 3 routing of traffic between the two VLANs, copy the following commands and paste them into the switch terminal window:



NOTE: The following example uses a version of Junos OS that supports Enhanced Layer 2 Software (ELS). When you use ELS, you create a Layer 3 virtual interface named **irb**. If you are using a version of Junos OS that does not support ELS, you create a Layer 3 virtual interface named **vlan**.

```
[edit]
set interfaces xe-0/0/4 unit 0 description "Sales server port"
set interfaces xe-0/0/4 unit 0 family ethernet-switching vlan members blue
set interfaces xe-0/0/6 unit 0 description "Sales wireless access point port"
set interfaces xe-0/0/6 unit 0 family ethernet-switching vlan members blue
set interfaces xe-0/0/0 unit 0 description "Support servers"
set interfaces xe-0/0/0 unit 0 family ethernet-switching vlan members red
set interfaces xe-0/0/2 unit 0 description "Support wireless access point port"
set interfaces xe-0/0/2 unit 0 family ethernet-switching vlan members red
set interfaces irb unit 100 family inet address 192.0.2.1/25
set interfaces irb unit 200 family inet address 192.0.2.129/25
set vlans blue l3-interface irb.100
set vlans blue vlan-id 100
set vlans red vlan-id 200
set vlans red l3-interface irb.200
```

Step-by-Step Procedure

To configure the switch interfaces and the VLANs to which they belong:

1. Configure the interface for the sales server in the blue VLAN:

```
[edit interfaces xe-0/0/4 unit 0]
user@switch# set description "Sales server port"
user@switch# set family ethernet-switching vlan members blue
```

2. Configure the interface for the wireless access point in the blue VLAN:

```
[edit interfaces xe-0/0/6 unit 0]
user@switch# set description "Sales wireless access point port"
user@switch# set family ethernet-switching vlan members blue
```

3. Configure the interface for the support server in the red VLAN:

```
[edit interfaces xe-0/0/0 unit 0]
user@switch# set description "Support server port"
user@switch# set family ethernet-switching vlan members red
```

4. Configure the interface for the wireless access point in the red VLAN:

```
[edit interfaces xe-0/0/2 unit 0]
user@switch# set description "Support wireless access point port"
user@switch# set family ethernet-switching vlan members red
```

Step-by-Step Procedure

Now create the VLANs and the IRB. The IRB will have logical units in the broadcast domains of both VLANs.

1. Create the red and blue VLANs by configuring the VLAN IDs for them:

```
[edit vlans]
user@switch# set blue vlan-id 100
user@switch# set red vlan-id 200
```

2. Create the interface named **irb** with a logical unit in the sales broadcast domain (blue VLAN):

```
[edit interfaces]
user@switch# set irb unit 100 family inet address 192.0.2.1/25
```

The unit number is arbitrary and does not have to match the VLAN tag ID. However, configuring the unit number to match the VLAN ID can help avoid confusion.

3. Add a logical unit in the support broadcast domain (red VLAN) to the **irb** interface:

```
[edit interfaces]
user@switch# set irb unit 200 family inet address 192.0.2.129/25
```

4. Complete the IRB configuration by binding the red and blue VLANs (Layer 2) with the appropriate logical units of the **irb** interface (Layer 3):

```
[edit vlans]
user@switch# set blue l3-interface irb.100
user@switch# set red l3-interface irb.200
```

Display the results of the configuration:

```
user@switch> show configuration
interfaces {
  xe-0/0/4 {
    unit 0 {
      description "Sales server port";
      family ethernet-switching {
        vlan members blue;
      }
    }
  }
  xe-0/0/6 {
    unit 0 {
      description "Sales wireless access point port";
      family ethernet-switching {
        vlan members blue;
      }
    }
  }
  xe-0/0/0 {
    unit 0 {
      description "Support server port";
      family ethernet-switching {
        vlan members red;
      }
    }
  }
  xe-0/0/2 {
    unit 0 {
      description "Support wireless access point port";
      family ethernet-switching {
        vlan members red;
      }
    }
  }
  irb {
    unit 100 {
      family inet address 192.0.2.1/25;
    }
    unit 200 {
      family inet address 192.0.2.129/25;
    }
  }
}
```

```

    }
  }
}
vllans {
  blue {
    vlan-id 100;
    interface xe-0/0/4.0;
    interface xe-0/0/6.0;
    l3-interface irb 100;
  }
  red {
    vlan-id 200;
    interface xe-0/0/0.0;
    interface xe-0/0/2.0;
    l3-interface irb 200;
  }
}
}

```



TIP: To quickly configure the blue and red VLAN interfaces, issue the `load merge terminal` command, copy the hierarchy, and paste it into the switch terminal window.

Verification

To verify that the **blue** and **red** VLANs have been created and are operating properly, perform these tasks:

- [Verifying That the VLANs Have Been Created and Associated with the Correct Interfaces on page 77](#)
- [Verifying That Traffic Can Be Routed Between the Two VLANs on page 78](#)

Verifying That the VLANs Have Been Created and Associated with the Correct Interfaces

Purpose Verify that the VLANs **blue** and **red** have been created on the switch and that all connected interfaces on the switch are members of the correct VLAN.

Action List all VLANs configured on the switch:

```

user@switch> show vlans
Name      Tag      Interfaces
default   100      xe-0/0/0.0, xe-0/0/2.0, xe-0/0/4.0, xe-0/0/6.0,
blue      100      xe-0/0/4.0, xe-0/0/6,
red       200      xe-0/0/0.0, xe-0/0/2.0, *
mgmt      me0.0*

```

Meaning The `show vlans` command lists all VLANs configured on the switch and which interfaces are members of each VLAN. This command output shows that the **blue** and **red** VLANs have been created. The **blue** VLAN has a tag ID of 100 and is associated with interfaces

xe-0/0/4.0 and **xe-0/0/6.0**. VLAN **red** has a tag ID of 200 and is associated with interfaces **xe-0/0/0.0** and **xe-0/0/2.0**.

Verifying That Traffic Can Be Routed Between the Two VLANs

Purpose Verify routing between the two VLANs.

Action Verify that the IRB logical units are up:

```
user@switch> show interfaces terse
irb.100                up    up    inet    192.0.2.1/25
irb.200                up    up    inet    192.0.2.129/25
```



NOTE: At least one port (access or trunk) with an appropriate VLAN assigned to it must be up for the irb interface to be up.

Verify that switch has created routes that use the IRB logical units:

```
user@switch> show route
192.0.2.0/25           *[Direct/0] 1d 03:26:45
                      > via irb.100
192.0.2.1/32           *[Local/0] 1d 03:26:45
                      Local via irb.100
192.0.2.128/25         *[Direct/0] 1d 03:26:45
                      > via irb.200
192.0.2.129/32         *[Local/0] 1d 03:26:45
                      Local via irb.200
```

List the Layer 3 routes in the switch's Address Resolution Protocol (ARP) table:

```
user@switch> show arp
MAC Address           Address           Name              Flags
00:00:0c:06:2c:0d     192.0.2.7        irb.100           None
00:13:e2:50:62:e0     192.0.2.132      irb.200           None
```

Meaning The output of the **show interfaces** and **show route** commands show that the Layer 3 IRB logical units are working and the switch has used them to create direct routes that it will use to forward traffic between the VLAN subnets. The **show arp** command displays the mappings between the IP addresses and MAC addresses for devices on both **irb.100** (associated with VLAN **blue**) and **irb.200** (associated with VLAN **red**). These two devices can communicate.

Related Documentation

- [Understanding Integrated Routing and Bridging on page 70](#)
- [irb \(Interfaces\) on page 291](#)
- [I3-interface on page 295](#)

Excluding a Routed VLAN Interface from State Calculations

Routed VLAN interfaces (RVIs) are used to bind specific VLANs to Layer 3 interfaces, enabling a switch to forward packets between those VLANs—without having to configure another device, such as a router, to connect VLANs. Because an RVI often has multiple ports in a single VLAN, the state calculation for a VLAN member might include a port that is down, possibly resulting in traffic loss.

Starting with Junos OS Release 14.1x53-D40 on QFX5100 switches, this feature enables you to exclude a trunk or access interface from the state calculation, which means that as soon as the port assigned to a member VLAN goes down, the RVI for the VLAN is also marked as down. In a typical scenario, one port on the interface is assigned to a single VLAN, while a second port on that interface is assigned to a trunk interface that carries traffic between multiple VLANs. A third port is often also assigned to an access interface to connect the VLAN to network devices.

Before you begin:

- Configure VLANs
- Configure RVIs for the VLANs.

For more information configuring RVIs, see [“Example: Configuring Routing Between VLANs on One Switch” on page 73](#).

To exclude an access or 802.1Q trunk interface from the state calculations for an RVI:

1. Configure a trunk or access interface.

```
[edit interfaces interface-name]
user@switch# set unit logical-unit-number family ethernet-switching port-mode
(access | trunk)
```

For example, configure interface xe-0/1/0.0 as a trunk interface:

```
[edit interfaces xe-0/1/0]
user@switch# set unit 0 family ethernet-switching port-mode trunk
```

2. Assign VLAN members to the access or trunk interface.

```
[edit interfaces interface-name unit logical-unit-number ethernet-switching]
user@switch# set vlan members [ (all | names | vlan-ids) ]
```

For example, assign all VLAN members configured on the device to the trunk interface xe-0/1/0:

```
[edit interfaces xe-0/1/0 unit 0 ethernet-switching]
user@switch# set vlan members all
```

3. Exclude an access or trunk interface from state calculations for the RVIs for member VLANs.

```
[edit interfaces interface-name ether-options]
user@switch# set autostate-exclude
```

For example, exclude the trunk interface xe-0/1/0 from state calculations for the RVIs for member VLANs:

```
[edit interfaces xe-0/1/0]
user@switch# set autostate-exclude
```

4. To confirm your configuration, from configuration mode, enter the **show interfaces xe-0/1/0** command. If your output does not display the intended configuration, repeat steps 1 through 4 to correct the configuration.

```
user@switch# show interfaces xe-0/1/0
ether-options {
  autostate-exclude;
}
unit 0 {
  family ethernet-switching {
    port-mode trunk;
    vlan {
      members all;
    }
  }
}
```

5. After you commit the configuration, issue the **show ethernet-switching interface xe-0/1/0.0** to verify that the logical interface is enabled with **autostate-exclude**.

```
user@switch> show ethernet-switching interface xe-0/1/0.0
Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
LH - MAC limit hit, DN - interface down,
SCTL - shutdown by Storm-control,
MMAS - Mac-move action shutdown, AS - Autostate-exclude enabled)
```

Logical interface	Vlan members	TAG	MAC limit	STP state	Logical interface flags	Tagging
xe-0/1/0.0	vlan_100	100	294912	Forwarding	AS	untagged

The **AS** in the **Logical interface flags** field indicates that **autostate-exclude** is enabled and that this interface will be excluded from the state calculations for the RVIs for the member VLANs.

Related Documentation

- [Understanding Integrated Routing and Bridging on page 70](#)

Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure)



NOTE: This task uses Junos OS for EX Series switches and Junos OS for QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure)*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

The Ethernet switching table, also known as the forwarding table, specifies the known locations of VLAN nodes and the addresses of devices within those nodes. There are two ways to populate the Ethernet switching table on a switch. The easiest method is to let the switch update the table with MAC addresses.

The second way to populate the Ethernet switching table is to manually insert addresses into the table. You can do this to reduce flooding and speed up the switch's automatic learning process.

Before configuring a static MAC address, be sure that you have:

- Set up the VLAN. See *Configuring VLANs for EX Series Switches (CLI Procedure)*.

To configure an interface to have a static MAC address:

```
[edit vlans vlan-name switch-options interface interface-name]
user@switch# set static-mac mac-address
```

Related Documentation

- *Understanding Bridging and VLANs on EX Series Switches*

Configuring Static ARP Entries

You can create static ARP table entries, which are explicit mappings between IP addresses and MAC addresses.

- To configure a static ARP entry:

```
[edit interfaces interface-name unit logical-unit-number family inet address
address]
user@switch# set arp ip-address l2-interface interface-name (mac | multicast-mac)
mac-address
```

The IP address that you specify must be part of the subnet defined in the enclosing **address** statement.

To associate a multicast MAC address with a unicast IP address, use the **multicast-mac** statement.

Specify the MAC address as 6 hexadecimal bytes in one of the following formats: *nnnn.nnnn.nnnn* or *nn:nn:nn:nn:nn:nn*; for example, 0011.2233.4455 or 00:11:22:33:44:55.



NOTE: You must use the `I2-interface` statement when you configure an IRB interface with a static ARP entry.

**Related
Documentation**

- [Understanding Static ARP Entries](#)
- [arp](#)

Understanding Multiple VLAN Registration Protocol (MVRP)

Multiple VLAN Registration Protocol (MVRP) is a Layer 2 messaging protocol that automates the creation and management of virtual LANs, thereby reducing the time you have to spend on these tasks. If your QFabric system connects to servers that host many virtual machines that require their own VLANs, using MVRP can save you the time and effort that would be required to manually create and administer the VLANs on the ports that connect to the servers. For example, if a virtual machine moves between servers—and therefore connects to a different redundant server Node group interface—MVRP can configure the appropriate VLAN membership on the new server Node group interface.

When using MVRP on a QFabric system, you must manually create on the QFabric the VLANs that exist on the attached servers because the QFabric implementation of MVRP does not allow VLANs to be created dynamically. However, you do not need to manually assign VLAN membership to the QFabric ports that connect to the servers. MVRP automatically assigns VLAN membership to server-facing QFabric ports when it learns about a VLAN from an attached server.

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



NOTE: MVRP on QFabric systems does not support private VLANs.

- [QFabric Requirements on page 82](#)
- [MVRP Operations on page 83](#)
- [MRP Timers Control MVRP Updates on page 83](#)
- [MVRP Uses MRP Messages to Transmit Switch and VLAN States on page 84](#)

QFabric Requirements

When configuring MVRP on a QFabric system, you can enable it globally or enable it only on the trunk ports that need to carry VLAN traffic from the attached servers. You also must manually create the expected VLANs, but you do not have to assign VLAN membership to the server-facing redundant server Node ports (as mentioned previously). However, you *do* have to manually assign VLAN membership to the uplink ports on the redundant server Node group and network Node group devices that will carry the VLAN

traffic. [Table 9 on page 83](#) summarizes the VLAN requirements for redundant server Node groups and network Node groups:

Table 9: MVRP VLAN Requirements for Node Devices

Node Group Type	Interface	Assign VLAN Membership to Trunk Ports?
Redundant server Node group	Server-facing trunk	No
Redundant server Node group	Uplink trunk (to interconnect device)	Yes
Network Node groups	Uplink trunk (to interconnect device)	Yes

MVRP Operations

MVRP stays synchronized by using MVRP protocol data units (PDUs). These PDUs specify which QFabric systems and switches are members of which VLANs, and which switch interfaces are in each VLAN. The MVRP PDUs are sent to other switches in the QFabric system when an MVRP state change occurs, and the receiving switches update their MVRP states accordingly. MVRP timers dictate when PDUs can be sent and when switches receiving MVRP PDUs can update their MVRP information.

In addition to this behavior, QFX switches include a mode—called passive mode—in which an MVRP-configured interface does not announce its membership in a VLAN or send any VLAN declarations (updates) unless it receives registration for that VLAN from a peer (server) on that interface. By default MVRP-configured interfaces behave in the standard manner and automatically send PDU updates to announce any VLAN changes. (This is called active mode.)

To enable passive mode on an interface, enter and commit this statement:

```
set protocols mvrp interface interface-name passive
```

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

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

MVRP is disabled by default and is valid only for trunk interfaces.

MRP Timers Control MVRP Updates

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

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

- Join timer—Controls the interval for the next MVRP PDU transmit opportunity.

- Leave timer—Controls the period of time that an interface on the switch waits in the leave state before changing to the unregistered state.
- LeaveAll timer—Controls the frequency with which the interface generates LeaveAll messages.



BEST PRACTICE: Unless there is a compelling reason to change the timer settings, leave the default settings in place. Modifying timers to inappropriate values can cause an imbalance in the operation of MVRP.

MVRP Uses MRP Messages to Transmit Switch and VLAN States

MVRP uses MRP messages to register and declare MVRP states for a interface or VLAN and to inform the switching network that a interface or VLAN is leaving MVRP. These messages are communicated in the MRP PDUs sent by MVRP-enabled interfaces.

The following MRP messages are communicated for MVRP:

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

Related Documentation

- [Understanding Bridging and VLANs on page 35](#)
- *Example: Configuring Automatic VLAN Administration Using MVRP*
- *Configuring Multiple VLAN Registration Protocol*

Troubleshooting Ethernet Switching

Problem **Description:** Sometimes a MAC address entry in the switch's Ethernet switching table is not updated after the device with that MAC address has been moved from one interface to another on the switch. Typically, the switch does not wait for a MAC address expiration when a MAC move operation occurs. As soon as the switch detects the MAC address on the new interface, it immediately updates the table. Many network devices send a gratuitous ARP packet when switching an IP address from one device to another. The switch updates its ARP cache table after receipt of such gratuitous ARP messages, and then it also updates its Ethernet switching table.

Sometimes silent devices, such as syslog servers or SNMP trap receivers that receive UDP traffic but do not return acknowledgment (ACK) messages to the traffic source, fail to send gratuitous ARP packets when a device moves. If such a move occurs when

the system administrator is not available to explicitly clear the affected interfaces by issuing the **clear ethernet-switching table** command, the entry for the moved device in the Ethernet switching table is not updated.

Solution Set up the switch to handle unattended MAC address switchovers.

1. Reduce the system-wide ARP aging timer. (By default, the ARP aging timer is set at 20 minutes. The range of the ARP aging timer is from 1 through 240 minutes.)

```
[edit system arp]
user@switch# set aging-timer 3
```

2. Set the MAC aging timer to the same value as the ARP timer. (By default, the MAC aging timer is set to 300 seconds. The range is 15 to 1,000,000 seconds.)

```
[edit vlans]
user@switch# set vlans sales mac-table-aging-time 180
```

The ARP entry and the MAC address entry for the moved device expire within the times specified by the aging timer values. After the entries expire, the switch sends a new ARP message to the IP address of the device. The device responds to the ARP message, thereby refreshing the entries in the switch's ARP cache table and Ethernet switching table.

- Related Documentation**
- [arp](#)
 - [mac-table-aging-time on page 318](#)

PART 3

MAC Addresses

- [Using MAC Addresses on page 89](#)

CHAPTER 3

Using MAC Addresses

- [Introduction to the Media Access Control \(MAC\) Layer 2 Sublayer on page 89](#)
- [Understanding MAC Learning on page 90](#)
- [Disabling MAC Learning on page 90](#)
- [Example: Disabling MAC Learning on page 91](#)
- [Configuring MAC Notification \(CLI Procedure\) on page 92](#)
- [Verifying That MAC Notification Is Working Properly on page 93](#)
- [Configuring MAC Limiting \(CLI Procedure\) on page 94](#)
- [Configuring MAC Table Aging on page 96](#)

Introduction to the Media Access Control (MAC) Layer 2 Sublayer

This topic provides an introduction to the MAC sublayer of the data link layer (Layer 2).

In Layer 2 of a network, the Media Access Control (MAC) sublayer provides addressing and channel access control mechanisms that enable several terminals or network nodes to communicate in a network.

The MAC sublayer acts as an interface between the logical link control (LLC) Ethernet sublayer and Layer 1 (the physical layer). The MAC sublayer emulates a full-duplex logical communication channel in a multipoint network. This channel may provide unicast, multicast, or broadcast communication service. The MAC sublayer uses MAC protocols to prevent collisions.

In Layer 2, multiple devices on the same physical link can uniquely identify one another at the data link layer, by using the MAC addresses that are assigned to all ports on a switch. A MAC algorithm accepts as input a secret key and an arbitrary-length message to be authenticated, and outputs a MAC address.

A MAC address is a 12-digit hexadecimal number (48 bits in long). MAC addresses are usually written in one of these formats:

- MM:MM:MM:SS:SS:SS
- MM-MM-MM-SS-SS-SS

The first half of a MAC address contains the ID number of the adapter manufacturer. These IDs are regulated by an Internet standards body. The second half of a MAC address represents the serial number assigned to the adapter by the manufacturer.

Contrast MAC addressing, which works at Layer 2, with IP addressing, which runs at Layer 3 (networking and routing). One way to remember the difference is that the MAC addresses apply to a physical or virtual node, whereas IP addresses apply to the software implementation of that node. MAC addresses are typically fixed on a per-node basis, whereas IP addresses change when the node moves from one part of the network to another.

IP networks maintain a mapping between the IP and MAC addresses of a node using the Address Resolution Protocol (ARP) table. DHCP also typically uses MAC addresses when assigning IP addresses to nodes.

**Related
Documentation**

- [Overview of Layer 2 Networking on page 31](#)
- [Understanding MAC Learning on page 90](#)

Understanding MAC Learning

MAC learning is the process of obtaining the MAC addresses of all the nodes on a network.

When a node is first connected to an Ethernet LAN or VLAN, it has no information about the other nodes on the network. As data is sent through the network, data packets include a data frame listing their source and destination MAC addresses. The data frame is forwarded to a target port, which is connected to the second device. The MAC address is learned locally at the target port, which facilitates communications for frames that later enter the target port and contain addresses previously learned from a received frame.

MAC learning can also be enabled on a per-VLAN basis. See *Example: Disabling MAC Learning in a VLAN* for further information.

By default, MAC learning is enabled on the QFX Series.

**Related
Documentation**

- [Introduction to the Media Access Control \(MAC\) Layer 2 Sublayer on page 89](#)
- [Overview of Layer 2 Networking on page 31](#)

Disabling MAC Learning

By default, MAC learning is globally enabled on all node. This topic describes how to disable MAC learning, as well as how to reenable and verify that MAC learning has been enabled or disabled.



NOTE: This task supports the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*. If your switch runs software that does not support ELS, see *Disabling MAC Learning*.

Disabling dynamic MAC learning prevents a node from learning source and destination MAC addresses.

- To disable MAC learning:

```
[edit vlans vlan-name switch-options interface interface-name]
user@switch# set no-mac-learning
```

- To enable MAC learning:

```
[edit vlans vlan-name switch-options interface interface-name]
user@switch# delete no-mac-learning
user@switch# deactivate no-mac-learning
```

- To verify the status of MAC learning, view the Ethernet MAC learning statistics in operational mode.

```
user@switch> show ethernet-switching table
Ethernet-switching table: 2 entries, 1 learned
  VLAN      MAC address      Type      Age Interfaces
  default   *                Flood     - All-members
  default   00:1f:12:39:90:80 Learn     29 xe-/0/0.0
```

Related Documentation

- [Understanding MAC Learning on page 90](#)
- [Example: Disabling MAC Learning on page 91](#)
- *no-mac-learning*

Example: Disabling MAC Learning

By default, MAC learning is enabled on the QFX Series. This topic provides examples for disabling, enabling, and verifying the operation of MAC learning on the QFX Series. These examples require that you be logged in as the root user to the switch on which you wish to modify MAC learning.



NOTE: This task uses Junos OS for QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Example: Disabling MAC Learning*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

- To disable MAC learning in a VLAN:

```
[edit]
user@switch# set vlans vlan10 switch-options interface xe-0/0/0.0 no-mac-learning
```

- To reenabling MAC learning:

```
[edit] vlans vlan10 switch-options interface xe-0/0/0.0
```

```
user@switch# delete no-mac-learning
```

- To verify the status of MAC learning on the QFX Series:

```
user@switch> show ethernet-switching table
```

```
Learning stats: 10 learn msg rcvd, 2 error, 0 forced update
```

Interface	Local pkts	Transit pkts	Error
xe-0/0/0.0	0	6	1
xe-0/0/22.0	0	0	0
xe-0/0/1.0	0	4	1
xe-0/0/2.0	0	0	0
xe-0/0/3.0	0	0	0
xe-0/0/4.0	0	0	0
xe-0/0/19.0	0	0	0
xe-0/0/18.0	0	0	0
xe-0/0/9.0	0	0	0

- Related Documentation**
- [Understanding MAC Learning on page 90](#)
 - [Disabling MAC Learning on page 90](#)
 - *no-mac-learning*

Configuring MAC Notification (CLI Procedure)



NOTE: This task uses the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Configuring MAC Notification (CLI Procedure)* or *Configuring MAC Notification*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

When a switch learns or unlearns a MAC address, SNMP notifications can be sent to the network management system at regular intervals to record the addition or removal of the MAC address. This process is known as MAC notification.

The MAC notification interval defines how often Simple Network Management Protocol (SNMP) notifications logging the addition or removal of MAC addresses on the switch are sent to the network management system.

MAC notification is disabled by default. When MAC notification is enabled, the default MAC notification interval is 30 seconds.

To enable or disable MAC notification, or to set the MAC notification interval, perform these tasks:

- [Enabling MAC Notification on page 93](#)
- [Disabling MAC Notification on page 93](#)
- [Setting the MAC Notification Interval on page 93](#)

Enabling MAC Notification

MAC notification is disabled by default. You need to perform this procedure to enable MAC notification.

To enable MAC notification on the switch with the default MAC notification interval of 30 seconds:

```
[edit switch-options]
user@switch# set mac-notification
```

To enable MAC notification on the switch with any other MAC notification interval (here, the MAC notification interval is set to 60 seconds):

```
[edit switch-options]
user@switch# set mac-notification notification-interval 60
```

Disabling MAC Notification

MAC notification is disabled by default. Perform this procedure only if MAC notification was previously enabled on your switch.

To disable MAC notification on the switch:

```
[edit switch-options]
user@switch# delete mac-notification
```

To disable MAC notification on a specific interface (here, the interface is ge-0/0/3):

```
[edit switch-options]
user@switch# set interface ge-0/0/3 no-mac-notification
```

Setting the MAC Notification Interval

The default MAC notification interval is 30 seconds. The procedure to change the MAC notification interval to a different interval is identical to the procedure to enable MAC notification on the switch with a nondefault value for the MAC notification interval.

To set the MAC notification interval on the switch (here, the MAC notification interval is set to 5 seconds):

```
[edit switch-options]
user@switch# set mac-notification notification-interval 5
```

Related Documentation

- *Verifying That MAC Notification Is Working Properly*

Verifying That MAC Notification Is Working Properly

Purpose Verify that MAC notification is enabled or disabled, and that the MAC notification interval is set to the specified value.

Action To verify that MAC notification is enabled or disabled and also to verify the MAC notification interval setting.

```
user@switch> show ethernet-switching mac-notification
Notification Status: Enabled
Notification Interval: 60
Notifications Sent      : 0
Notifications Table Maxsize : 256
```

Meaning The output in the **Notification Status** field shows that MAC notification is enabled. The output in the **Notification Status** field would display **Disabled** if MAC notification was disabled.

The **Notification Interval** field output shows that the MAC notification interval is set to 60 seconds.

Related Documentation

- [Configuring MAC Notification](#)
- [Configuring MAC Notification \(CLI Procedure\) on page 92](#)

Configuring MAC Limiting (CLI Procedure)



NOTE: This task uses Junos OS for EX Series switches and QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Configuring MAC Limiting (CLI Procedure)*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

This topic describes various ways of configuring a limitation on MAC addresses in packets that are received and forwarded by the switch.



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

The different ways of setting a MAC limit are described in the following sections:

- [Limiting the Number of MAC Addresses Learned by an Interface on page 95](#)
- [Limiting the Number of MAC Addresses Learned by a VLAN on page 95](#)

Limiting the Number of MAC Addresses Learned by an Interface

To secure a port, you can set the maximum number of MAC addresses that can be learned by an interface:

- Set the MAC limit on an interface, and specify an action that the switch takes after the specified limit is exceeded:

```
[edit switch-options]
user@switch# set interface interface-name interface-mac-limit limit packet-action
action
```

After you set a new MAC limit for the interface, the system clears existing entries in the MAC address forwarding table associated with the interface.

Limiting the Number of MAC Addresses Learned by a VLAN

To limit the number of MAC addresses learned by a VLAN, perform both of the following steps:

- Set the maximum number of MAC addresses that can be learned by a VLAN, and specify an action that the switch takes after the specified limit is exceeded:

```
[edit vlans]
user@switch# set vlan-name switch-options mac-table-size limit packet-action
action
```

- Set the maximum number of MAC addresses that can be learned by one or all interfaces in the VLAN, and specify an action that the switch takes after the specified limit is exceeded:

```
[edit vlans]
user@switch# set vlan-name switch-options interface interface-name
interface-mac-limit limit packet-action action
[edit vlans]
user@switch# set vlan-name switch-options interface-mac-limit limit packet-action
action
```



NOTE: If you specify a MAC limit and packet action for all interfaces in the VLAN *and* a specific interface in the VLAN, the MAC limit and packet action specified at the specific interface level takes precedence. Also, at the VLAN interface level, only the drop and drop-and-log options are supported.

After you set new MAC limits for a VLAN by using the **mac-table-size** statement or for interfaces associated with a VLAN by using the **interface-mac-limit** statement, the system clears the corresponding existing entries in the MAC address forwarding table.

- Related Documentation**
- Understanding Bridging and VLANs on EX Series Switches*
 - Configuring Persistent MAC Learning (CLI Procedure)*

Configuring MAC Table Aging

MAC table aging ensures that a switch tracks only active nodes on the network and that it is able to flush out network nodes that are no longer available.

To manage MAC entries more efficiently, you can configure an entry's aging time, which is the maximum time that an entry can remain in the MAC address table before it is deleted because it has reached its maximum age.



NOTE: This task uses Junos OS for QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Configuring MAC Table Aging*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

You can use the **set-mac-table-aging-time** command to configure how long entries remain in the Ethernet switching table before expiring. Here the VLAN is **employee-vlan**:

```
[edit vlans employee-vlan switch-options]
user@switch# set mac-table-aging-time 200
```

Related Documentation

- [Understanding Bridging and VLANs on page 35](#)
- [Example: Setting Up Bridging with Multiple VLANs on page 60](#)
- [Example: Connecting an Access Switch to a Distribution Switch](#)

PART 4

Spanning Trees

- [Using Spanning Trees on page 99](#)

CHAPTER 4

Using Spanning Trees

- Overview of Spanning-Tree Protocols on page 100
- Understanding MSTP on page 101
- Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101
- Understanding RSTP on page 121
- Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123
- Configuring RSTP (CLI Procedure) on page 139
- Understanding VSTP on page 140
- Example: Configuring VSTP on QFX Series Switches and EX4600 Switches on page 141
- Understanding BPDU Protection for STP, RSTP, and MSTP on page 146
- Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations on EX Series Switches on page 147
- Understanding Loop Protection for STP, RSTP, VSTP, and MSTP on page 151
- Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on page 152
- Understanding Root Protection for STP, RSTP, VSTP, and MSTP on page 156
- Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees on page 157

Overview of Spanning-Tree Protocols

QFX Series switches provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), Multiple Spanning Tree Protocol (MSTP), and VLAN Spanning Tree Protocol (VSTP). The default spanning-tree protocol on the QFX Series is RSTP. RSTP provides faster convergence times than STP. However, some legacy networks require the slower convergence times of basic STP.

The STP support provided for the QFX Series includes:

- IEEE 802.1d
- 802.1w RSTP
- 802.1s MSTP

If your network includes IEEE 802.1D 1998 bridges, you can remove RSTP and explicitly configure STP. When you explicitly configure STP, the QFX Series products use the IEEE 802.1D 2004 specification, force version 0. This configuration runs a version of RSTP that is compatible with the classic, basic STP. If you use virtual LANs (VLANs), you should enable VSTP and use it on your network. See [“Understanding VSTP” on page 140](#).

You can use the same operational commands (**show spanning-tree bridge** and **show spanning-tree interface**) to check the status of your spanning-tree configuration, regardless of which spanning-tree protocol has been configured.

STP uses bridge protocol data unit (BPDU) packets to exchange information with other switches. BPDUs send hello packets out at regular intervals to exchange information across bridges and detect loops in a network topology. There are two types of BPDUs:

- Configuration BPDUs—These BPDUs contain configuration information about the transmitting switch and its ports, including switch and port MAC addresses, switch priority, port priority, and port cost.
- Topology change notification (TCN) BPDUs—When a bridge needs to signal a topology change, it starts to send TCNs on its root port. The designated bridge receives the TCN, acknowledges it, and generates another one for its own root port. The process continues until the TCN reaches the root bridge.

STP uses the information provided by the BPDUs to elect a root bridge, identify root ports for each switch, identify designated ports for each physical LAN segment, and prune specific redundant links to create a loop-free tree topology. All leaf devices calculate the best path to the root device and place their ports in blocking or forwarding states based on the best path to the root. The resulting tree topology provides a single active Layer 2 data path between any two end stations.

Understanding Spanning Tree Protocols on a QFabric System

Although there is no need to run STP in a QFabric system, you can connect a QFabric system to another Layer 2 device and use STP. STP traffic can only be processed on network Node groups. Other Node groups, such as redundant server Node groups and server Node groups, discard the STP bridge protocol data units (BPDUs) traffic and disable the

interface automatically. Server Node groups only process host-facing protocols, whereas Network Node groups process all supported protocols.

**Related
Documentation**

- [Understanding BPDU Protection for STP, RSTP, and MSTP on page 146](#)
- [Understanding MSTP on page 101](#)
- [Understanding RSTP on page 121](#)
- [Understanding VSTP on page 140](#)

Understanding MSTP

Although RSTP provides faster convergence time than STP does, it still does not solve a problem inherent in STP: all VLANs within a LAN must share the same spanning tree. To solve this problem, the QFX Series products use Multiple Spanning Tree Protocol (MSTP) to create a loop-free topology in networks with multiple spanning-tree regions.

An MSTP region allows a group of bridges to be modeled as a single bridge. An MSTP region contains multiple spanning-tree instances (MSTIs). MSTIs provide different paths for different VLANs. This functionality facilitates more efficient load sharing across redundant links.

An MSTP region can support up to 64 MSTIs, and each instance can support from 1 through 4094 VLANs.

**Related
Documentation**

- [Overview of Spanning-Tree Protocols on page 100](#)
- [Understanding RSTP on page 121](#)
- [Example: Configuring Network Regions for VLANs with MSTP](#)

Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches



NOTE: This example uses Junos OS for EX Series switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

Multiple Spanning Tree Protocol (MSTP) is used to create a loop-free topology in networks using multiple spanning-tree regions in which each region contains multiple spanning-tree instances (MSTIs). MSTIs provide different paths for different VLANs. This functionality facilitates better load sharing across redundant links.

Up to 64 MSTIs can be created for an EX Series switch, and each MSTI can support up to 4094 VLANs.

This example describes how to configure MSTP on four EX Series switches:

- [Requirements on page 102](#)
- [Overview and Topology on page 102](#)
- [Configuring MSTP on Switch 1 on page 105](#)
- [Configuring MSTP on Switch 2 on page 108](#)
- [Configuring MSTP on Switch 3 on page 110](#)
- [Configuring MSTP on Switch 4 on page 113](#)
- [Verification on page 115](#)

Requirements

This example uses the following hardware and software components:

- Junos OS Release 13.2X50-D10 or later or later for EX Series switches
- Four EX Series switches

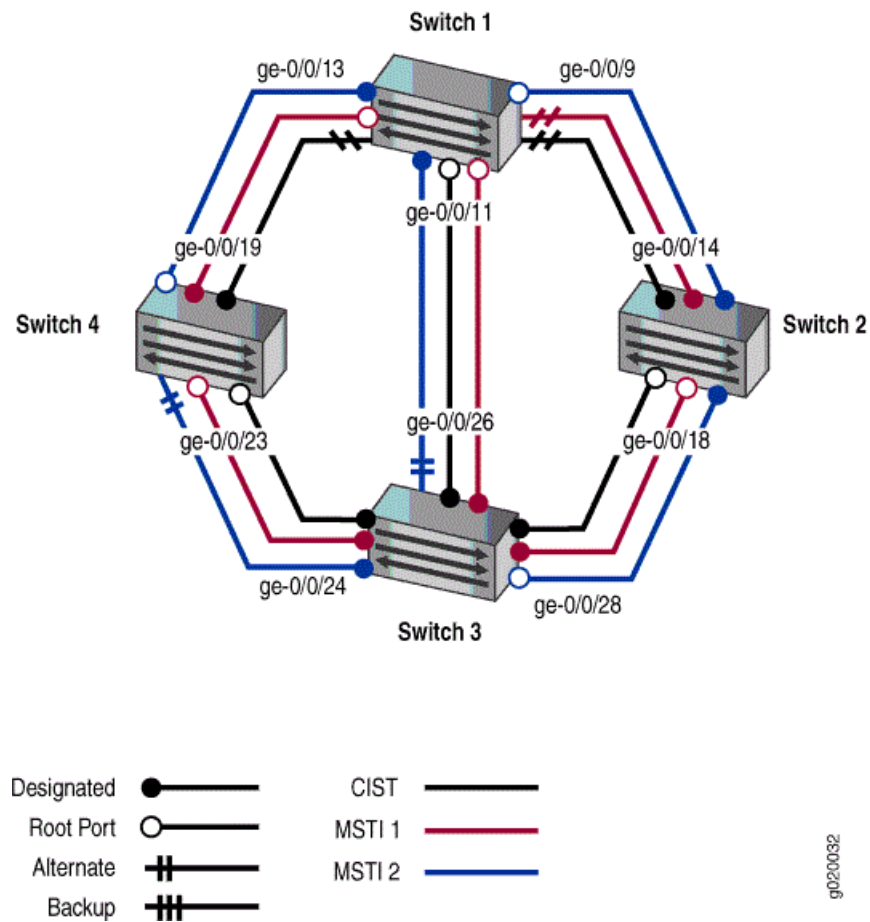
Before you configure the switches for MSTP, be sure you have:

- Installed and connected the four switches. See the hardware documentation for your switch.
- Performed the initial software configuration on all switches. See *Connecting and Configuring an EX Series Switch (CLI Procedure)* or *Connecting and Configuring an EX Series Switch (J-Web Procedure)*.

Overview and Topology

When the number of VLANs grows in a network, MSTP provides an efficient way of creating a loop-free topology by using MSTIs. Each MSTI in the spanning-tree domain maintains its own tree. Each tree can be mapped to different links, utilizing bandwidth that would be unavailable to a single tree. MSTIs reduce the demand on system resources.

Figure 5: Network Topology for MSTP



The interfaces shown in [Figure 5 on page 103](#) will be configured for MSTP.



NOTE: You can configure MSTP only on physical interfaces, not on logical interfaces.

Table 10: Components of the Topology for Configuring MSTP on EX Series Switches

Property	Settings
Switch 1	<p>The following interfaces on Switch 1 are connected in this way:</p> <ul style="list-style-type: none"> • ge-0/0/9 is connected to Switch 2 • ge-0/0/13 is connected to Switch 4 • ge-0/0/11 is connected to Switch 3
Switch 2	<p>The following interfaces on Switch 2 are connected in this way:</p> <ul style="list-style-type: none"> • ge-0/0/14 is connected to Switch 1 • ge-0/0/18 is connected to Switch 3

Table 10: Components of the Topology for Configuring MSTP on EX Series Switches (*continued*)

Property	Settings
Switch 3	<p>The following interfaces on Switch 3 are connected in this way:</p> <ul style="list-style-type: none"> • ge-0/0/26 is connected to Switch 1 • ge-0/0/28 is connected to Switch 2 • ge-0/0/24 is connected to Switch 4
Switch 4	<p>The following interfaces on Switch 4 are connected in this way:</p> <ul style="list-style-type: none"> • ge-0/0/19 is connected to Switch 1 • ge-0/0/23 is connected to Switch 3
VLAN names and tag IDs	voice-vlan , tag 10 employee-vlan , tag 20 guest-vlan , tag 30 camera-vlan , tag 40
MSTIs	1 2
MSTI region	region1

The topology in [Figure 5 on page 103](#) shows a common and internal spanning tree (CIST). The CIST is a single spanning tree connecting all devices in the network. The switch with the lowest bridge priority is elected as the root bridge of the CIST. You can control the election of the root bridge by configuring the bridge priority. Switch 3 is the root bridge of the CIST.

The ports in an MSTP topology have specific roles:

- The *root port* is responsible for forwarding data to the root bridge.
- The *alternate port* is a standby port for the root port. When a root port goes down, the alternate port becomes the active root port.
- The *designated port* forwards data to the downstream network segment or device.
- The *backup port* becomes the active designated port and starts forwarding data when the designated port goes down.

In this example, one MSTP region contains Switch 1, Switch 2, Switch 3, and Switch 4. Within the region, four VLANs are created:

- **voice-vlan** supports voice traffic and has the VLAN tag identifier of **10**.
- **employee-vlan** supports data traffic and has the VLAN tag identifier of **20**.
- **guest-vlan** supports guest VLAN traffic (for supplicants that fail authentication) and has the VLAN tag identifier of **30**.
- **camera-vlan** supports video traffic and has the VLAN tag identifier of **40**.

The VLANs are associated with specific interfaces on each of the four switches. Two MSTIs, 1 and 2, are then associated with the VLAN tag identifiers, and some MSTP parameters, such as cost, are configured on each switch.

Configuring MSTP on Switch 1

CLI Quick Configuration To quickly configure interfaces and MSTP on Switch 1, copy the following commands and paste them into the switch terminal window:

```
[edit]
set vlans voice-vlan description "Voice VLAN"
set vlans voice-vlan vlan-id 10
set vlans employee-vlan description "Employee VLAN"
set vlans employee-vlan vlan-id 20
set vlans guest-vlan description "Guest VLAN"
set vlans guest-vlan vlan-id 30
set vlans camera-vlan description "Camera VLAN"
set vlans camera-vlan vlan-id 40
set interfaces ge-0/0/13 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/9 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/13 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/9 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk
set protocols mstp configuration-name region1
set protocols mstp bridge-priority 16k
set protocols mstp interface ge-0/0/13 cost 1000
set protocols mstp interface ge-0/0/13 mode point-to-point
set protocols mstp interface ge-0/0/9 cost 1000
set protocols mstp interface ge-0/0/9 mode point-to-point
set protocols mstp interface ge-0/0/11 cost 1000
set protocols mstp interface ge-0/0/11 mode point-to-point
set protocols mstp msti 1 bridge-priority 16k
set protocols mstp msti 1 vlan [10 20]
set protocols mstp msti 1 interface ge-0/0/11 cost 1000
set protocols mstp msti 2 bridge-priority 8k
set protocols mstp msti 2 vlan [30 40]
```

Step-by-Step Procedure To configure interfaces and MSTP on Switch 1:

1. Configure the VLANs `voice-vlan`, `employee-vlan`, `guest-vlan`, and `camera-vlan`:

```
[edit vlans]
user@switch1# set voice-vlan description "Voice VLAN"
user@switch1# set voice-vlan vlan-id 10
user@switch1# set employee-vlan description "Employee VLAN"
user@switch1# set employee-vlan vlan-id 20
user@switch1# set guest-vlan description "Guest VLAN"
user@switch1# set guest-vlan vlan-id 30
user@switch1# set camera-vlan description "Camera VLAN"
user@switch1# set guest-vlan vlan-id 40
```

2. Configure the VLANs on the interfaces, including support for the Ethernet Switching protocol:

```
[edit interfaces]
user@switch1# set ge-0/0/13 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch1# set ge-0/0/9 unit 0 family ethernet-switching vlan members [10 20 30 40]
```

- ```

user@switch1# set ge-0/0/11 unit 0 family ethernet-switching vlan members [10 20 30 40]

```
3. Configure the port mode for the interfaces:

```

[edit interfaces]
user@switch1# set ge-0/0/13 unit 0 family ethernet-switching interface-mode trunk
user@switch1# set ge-0/0/9 unit 0 family ethernet-switching interface-mode trunk
user@switch1# set ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk

```
  4. Configure MSTP on the switch, including the two MSTIs:

```

[edit protocols]
user@switch1# mstp configuration-name region1
user@switch1# mstp bridge-priority 16k
user@switch1# mstp interface ge-0/0/13 cost 1000
user@switch1# mstp interface ge-0/0/13 mode point-to-point
user@switch1# mstp interface ge-0/0/9 cost 1000
user@switch1# mstp interface ge-0/0/9 mode point-to-point
user@switch1# mstp interface ge-0/0/11 cost 1000
user@switch1# mstp interface ge-0/0/11 mode point-to-point
user@switch1# mstp msti 1 bridge-priority 16k
user@switch1# mstp msti 1 vlan [10 20]
user@switch1# mstp msti 1 interface ge-0/0/11 cost 1000
user@switch1# mstp msti 2 bridge-priority 8k
user@switch1# mstp msti 2 vlan [30 40]

```

**Results** Check the results of the configuration:

```

user@switch1> show configuration
interfaces {
 ge-0/0/13 {
 unit 0 {
 family ethernet-switching {
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
 }
 }
 ge-0/0/9 {
 unit 0 {
 family ethernet-switching {
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
 }
 }
 ge-0/0/11 {
 unit 0 {
 family ethernet-switching {

```



```
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
}
}
protocols {
 mstp {
 configuration-name region1;
 bridge-priority 16k;
 interface ge-0/0/13 {
 cost 1000;
 mode point-to-point;
 }
 interface ge-0/0/9 {
 cost 1000;
 mode point-to-point;
 }
 interface ge-0/0/11 {
 cost 1000;
 mode point-to-point;
 }
 }
 msti 1 {
 bridge-priority 16k;
 vlan [10 20];
 interface ge-0/0/11 {
 cost 1000;
 }
 }
 msti 2 {
 bridge-priority 8k;
 vlan [30 40];
 }
}
vlangs {
 voice-vlan {
 vlan-id 10;
 }
 employee-vlan {
 vlan-id 20;
 }
 guest-vlan {
 vlan-id 30;
 }
 camera-vlan {
 vlan-id 40;
 }
}
```

## Configuring MSTP on Switch 2

**CLI Quick Configuration** To quickly configure interfaces and MSTP on Switch 2, copy the following commands and paste them into the switch terminal window:

```
[edit]
set vlans voice-vlan description "Voice VLAN"
set vlans voice-vlan vlan-id 10
set vlans employee-vlan description "Employee VLAN"
set vlans employee-vlan vlan-id 20
set vlans guest-vlan description "Guest VLAN"
set vlans guest-vlan vlan-id 30
set vlans camera-vlan description "Camera VLAN"
set vlans camera-vlan vlan-id 40
set interfaces ge-0/0/14 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/18 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/18 unit 0 family ethernet-switching interface-mode trunk
set protocols mstp configuration-name region1
set protocols mstp bridge-priority 32k
set protocols mstp interface ge-0/0/14 cost 1000
set protocols mstp interface ge-0/0/14 mode point-to-point
set protocols mstp interface ge-0/0/18 cost 1000
set protocols mstp interface ge-0/0/18 mode point-to-point
set protocols mstp msti 1 bridge-priority 32k
set protocols mstp msti 1 vlan [10 20]
set protocols mstp msti 2 bridge-priority 4k
set protocols mstp msti 2 vlan [30 40]
```

**Step-by-Step Procedure** To configure interfaces and MSTP on Switch 2:

1. Configure the VLANs **voice-vlan**, **employee-vlan**, **guest-vlan**, and **camera-vlan**:

```
[edit vlans]
user@switch2# set voice-vlan description "Voice VLAN"
user@switch2# set voice-vlan vlan-id 10
user@switch2# set employee-vlan description "Employee VLAN"
user@switch2# set employee-vlan vlan-id 20
user@switch2# set guest-vlan description "Guest VLAN"
user@switch2# set guest-vlan vlan-id 30
user@switch2# set camera-vlan vlan-description "Camera VLAN"
user@switch2# set guest-vlan vlan-id 40
```

2. Configure the VLANs on the interfaces, including support for the Ethernet Switching protocol:

```
[edit interfaces]
user@switch2# set ge-0/0/14 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch2# set ge-0/0/18 unit 0 family ethernet-switching vlan members [10 20 30 40]
```

3. Configure the port mode for the interfaces:

```
[edit interfaces]
user@switch2# set ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk
user@switch2# set ge-0/0/18 unit 0 family ethernet-switching interface-mode trunk
```

4. Configure MSTP on the switch, including the two MSTIs:

```
[edit protocols]
user@switch2# mstp configuration-name region1
user@switch2# mstp bridge-priority 32k
```

```

user@switch2# mstp interface ge-0/0/14 cost 1000
user@switch2# mstp interface ge-0/0/14 mode point-to-point
user@switch2# mstp interface ge-0/0/18 cost 1000
user@switch2# mstp interface ge-0/0/18 mode point-to-point
user@switch2# mstp msti 1 bridge-priority 32k
user@switch2# mstp msti 1 vlan [10 20]
user@switch2# mstp msti 2 bridge-priority 4k
user@switch2# mstp msti 2 vlan [30 40]

```

**Results** Check the results of the configuration:

```

user@switch2> show configuration
interfaces {
 ge-0/0/14 {
 unit 0 {
 family ethernet-switching {
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
 }
 }
 ge-0/0/18 {
 unit 0 {
 family ethernet-switching {
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
 }
 }
}
protocols {
 mstp {
 configuration-name region1;
 bridge-priority 32k;
 interface ge-0/0/14 {
 cost 1000;
 mode point-to-point;
 }
 interface ge-0/0/18 {
 cost 1000;
 mode point-to-point;
 }
 msti 1 {
 bridge-priority 32k;
 vlan [10 20];
 }
 }
}

```

```
msti 2 {
 bridge-priority 4k;
 vlan [30 40];
}
}
vlangs {
 voice-vlan {
 vlan-id 10;
 }
 employee-vlan {
 vlan-id 20;
 }
 guest-vlan {
 vlan-id 30;
 }
 camera-vlan {
 vlan-id 40;
 }
}
```

### Configuring MSTP on Switch 3

**CLI Quick Configuration** To quickly configure interfaces and MSTP on Switch 3, copy the following commands and paste them into the switch terminal window:

```
[edit]
set vlans voice-vlan description "Voice VLAN"
set vlans voice-vlan vlan-id 10
set vlans employee-vlan description "Employee VLAN"
set vlans employee-vlan vlan-id 20
set vlans guest-vlan description "Guest VLAN"
set vlans guest-vlan vlan-id 30
set vlans camera-vlan description "Camera VLAN"
set vlans camera-vlan vlan-id 40
set interfaces ge-0/0/26 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/28 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/24 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/26 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/28 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/24 unit 0 family ethernet-switching interface-mode trunk
set protocols mstp configuration-name region1
set protocols mstp bridge-priority 8k
set protocols mstp interface ge-0/0/26 cost 1000
set protocols mstp interface ge-0/0/26 mode point-to-point
set protocols mstp interface ge-0/0/28 cost 1000
set protocols mstp interface ge-0/0/28 mode point-to-point
set protocols mstp interface ge-0/0/24 cost 1000
set protocols mstp interface ge-0/0/24 mode point-to-point
set protocols mstp msti 1 bridge-priority 4k
set protocols mstp msti 1 vlan [10 20]
set protocols mstp msti 2 bridge-priority 16k
set protocols mstp msti 2 vlan [30 40]
```

**Step-by-Step Procedure** To configure interfaces and MSTP on Switch 3:

1. Configure the VLANs voice-vlan, employee-vlan, guest-vlan, and camera-vlan:

```
[edit vlans]
user@switch3# set voice-vlan description "Voice VLAN"
user@switch3# set voice-vlan vlan-id 10
user@switch3# set employee-vlan description "Employee VLAN"
user@switch3# set employee-vlan vlan-id 20
user@switch3# set guest-vlan description "Guest VLAN"
user@switch3# set guest-vlan vlan-id 30
user@switch3# set camera-vlan description "Camera VLAN"
user@switch3# set guest-vlan vlan-id 40
```

2. Configure the VLANs on the interfaces, including support for the Ethernet Switching protocol:

```
[edit interfaces]
user@switch3# set ge-0/0/26 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch3# set ge-0/0/28 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch3# set ge-0/0/24 unit 0 family ethernet-switching vlan members [10 20 30 40]
```

3. Configure the port mode for the interfaces:

```
[edit interfaces]
user@switch3# set ge-0/0/26 unit 0 family ethernet-switching interface-mode trunk
user@switch3# set ge-0/0/28 unit 0 family ethernet-switching interface-mode trunk
user@switch3# set ge-0/0/24 unit 0 family ethernet-switching interface-mode trunk
```

4. Configure MSTP on the switch, including the two MSTIs:

```
[edit protocols]
user@switch3# mstp configuration-name region1
user@switch3# mstp bridge-priority 8k
user@switch3# mstp interface ge-0/0/26 cost 1000
user@switch3# mstp interface ge-0/0/26 mode point-to-point
user@switch3# mstp interface ge-0/0/28 cost 1000
user@switch3# mstp interface ge-0/0/28 mode point-to-point
user@switch3# mstp interface ge-0/0/24 cost 1000
user@switch3# mstp interface ge-0/0/24 mode point-to-point
user@switch3# mstp msti 1 bridge-priority 4k
user@switch3# mstp msti 1 vlan [10 20]
user@switch3# mstp msti 2 bridge-priority 16k
user@switch3# mstp msti 2 vlan [30 40]
```

**Results** Check the results of the configuration:

```
user@switch3> show configuration
interfaces {
 ge-0/0/26 {
 unit 0 {
 family ethernet-switching {
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
 }
 }
}
```

Copyright © 2016, Juniper Networks, Inc.

```

}
vlands {
 voice-vlan {
 vlan-id 10;
 }
 employee-vlan {
 vlan-id 20;
 }
 guest-vlan {
 vlan-id 30;
 }
 camera-vlan {
 vlan-id 40;
 }
}

```

### Configuring MSTP on Switch 4

**CLI Quick Configuration** To quickly configure interfaces and MSTP on Switch 4, copy the following commands and paste them into the switch terminal window:

```

[edit]
set vlans voice-vlan description "Voice VLAN"
set vlans voice-vlan vlan-id 10
set vlans employee-vlan description "Employee VLAN"
set vlans employee-vlan vlan-id 20
set vlans guest-vlan description "Guest VLAN"
set vlans guest-vlan vlan-id 30
set vlans camera-vlan description "Camera VLAN"
set vlans camera-vlan vlan-id 40
set interfaces ge-0/0/23 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/19 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/23 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/19 unit 0 family ethernet-switching interface-mode trunk
set protocols mstp configuration-name region1
set protocols mstp bridge-priority 16k
set protocols mstp interface ge-0/0/23 cost 1000
set protocols mstp interface ge-0/0/23 mode point-to-point
set protocols mstp interface ge-0/0/19 cost 1000
set protocols mstp interface ge-0/0/19 mode point-to-point
set protocols mstp msti 1 bridge-priority 16k
set protocols mstp msti 1 vlan [10 20]
set protocols mstp msti 2 bridge-priority 32k
set protocols mstp msti 2 vlan [30 40]

```

**Step-by-Step Procedure** To configure interfaces and MSTP on Switch 4:

1. Configure the VLANs `voice-vlan`, `employee-vlan`, `guest-vlan`, and `camera-vlan`:

```

[edit vlans]
user@switch4# set voice-vlan description "Voice VLAN"
user@switch4# set voice-vlan vlan-id 10
user@switch4# set employee-vlan description "Employee VLAN"
user@switch4# set employee-vlan vlan-id 20
user@switch4# set guest-vlan description "Guest VLAN"
user@switch4# set guest-vlan vlan-id 30
user@switch4# set camera-vlan description "Camera VLAN"
user@switch4# set camera-vlan vlan-id 40

```

2. Configure the VLANs on the interfaces, including support for the Ethernet Switching protocol:

```
[edit interfaces]
user@switch4# set ge-0/0/23 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch4# set ge-0/0/19 unit 0 family ethernet-switching vlan members [10 20 30 40]
```

3. Configure the port mode for the interfaces:

```
[edit interfaces]
user@switch4# set ge-0/0/23 unit 0 family ethernet-switching interface-mode trunk
user@switch4# set ge-0/0/19 unit 0 family ethernet-switching interface-mode trunk
```

4. Configure MSTP on the switch, including the two MSTIs:

```
[edit protocols]
user@switch4# mstp configuration-name region1
user@switch4# mstp bridge-priority 16k
user@switch4# mstp interface ge-0/0/23 cost 1000
user@switch4# mstp interface ge-0/0/23 mode point-to-point
user@switch4# mstp interface ge-0/0/19 cost 1000
user@switch4# mstp interface ge-0/0/19 mode point-to-point
user@switch4# mstp msti 1 bridge-priority 16k
user@switch4# mstp msti 1 vlan [10 20]
user@switch4# mstp msti 2 bridge-priority 32k
user@switch4# mstp msti 2 vlan [30 40]
```

**Results** Check the results of the configuration:

```
user@switch4> show configuration
interfaces {
 ge-0/0/23 {
 unit 0 {
 family ethernet-switching {
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
 }
 }
 ge-0/0/19 {
 unit 0 {
 family ethernet-switching {
 interface-mode trunk;
 vlan {
 members 10;
 members 20;
 members 30;
 members 40;
 }
 }
 }
 }
}
```



```

protocols {
 mstp {
 configuration-name region1;
 bridge-priority 16k;
 interface ge-0/0/23 {
 cost 1000;
 mode point-to-point;
 }
 interface ge-0/0/19 {
 cost 1000;
 mode point-to-point;
 }
 msti 1 {
 bridge-priority 16k;
 vlan [10 20];
 }
 msti 2 {
 bridge-priority 32k;
 vlan [30 40];
 }
 }
}
vlands {
 voice-vlan {
 vlan-id 10;
 }
 employee-vlan {
 vlan-id 20;
 }
 guest-vlan {
 vlan-id 30;
 }
 camera-vlan {
 vlan-id 40;
 }
}
}

```

## Verification

To confirm that the configuration is working properly, perform these tasks:

- [Verifying MSTP Configuration on Switch 1 on page 115](#)
- [Verifying MSTP Configuration on Switch 2 on page 117](#)
- [Verifying MSTP Configuration on Switch 3 on page 118](#)
- [Verifying MSTP Configuration on Switch 4 on page 120](#)

### Verifying MSTP Configuration on Switch 1

**Purpose** Verify the MSTP configuration on Switch 1.

**Action** Issue the operational mode commands **show spanning-tree interface** and **show spanning-tree bridge**:

```
user@switch1> show spanning-tree interface
```

## Spanning tree interface parameters for instance 0

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/9  | 128:490 | 128:490               | 16384.4c9614e9f841      | 1000         | BLK   | DIS  |
| ge-0/0/11 | 128:491 | 128:491               | 16384.4c9614e9f841      | 1000         | BLK   | DIS  |
| ge-0/0/13 | 128:492 | 128:492               | 16384.4c9614e9f841      | 1000         | BLK   | DIS  |

## Spanning tree interface parameters for instance 1

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/9  | 128:490 | 128:490               | 16385.4c9614e9f841      | 1000         | BLK   | DIS  |
| ge-0/0/11 | 128:491 | 128:491               | 16385.4c9614e9f841      | 1000         | BLK   | DIS  |
| ge-0/0/13 | 128:492 | 128:492               | 16385.4c9614e9f841      | 1000         | BLK   | DIS  |

## Spanning tree interface parameters for instance 2

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/9  | 128:490 | 128:490               | 8194.4c9614e9f841       | 1000         | BLK   | DIS  |
| ge-0/0/11 | 128:491 | 128:491               | 8194.4c9614e9f841       | 1000         | BLK   | DIS  |
| ge-0/0/13 | 128:492 | 128:492               | 8194.4c9614e9f841       | 1000         | BLK   | DIS  |

```
user@switch1> show spanning-tree bridge
```

## STP bridge parameters

```
Routing instance name : GLOBAL
Context ID : 0
Enabled protocol : MSTP
```

## STP bridge parameters for CIST

```
Root ID : 16384.4c:96:14:e9:f8:41
CIST regional root : 16384.4c:96:14:e9:f8:41
CIST internal root cost : 0
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Number of topology changes : 0
Local parameters
 Bridge ID : 16384.4c:96:14:e9:f8:41
```

## STP bridge parameters for MSTI 1

```
MSTI regional root : 16385.4c:96:14:e9:f8:41
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Number of topology changes : 0
Local parameters
 Bridge ID : 16385.4c:96:14:e9:f8:41
```

## STP bridge parameters for MSTI 2

```
MSTI regional root : 8194.4c:96:14:e9:f8:41
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Number of topology changes : 0
Local parameters
 Bridge ID : 8194.4c:96:14:e9:f8:41
```

**Meaning** The operational mode command **show spanning-tree interface** displays spanning-tree domain information such as the designated port and the port roles.

The operational mode command **show spanning-tree bridge** displays the spanning-tree domain information at either the bridge level or the interface level. If the optional interface name is omitted, all interfaces in the spanning-tree domain are displayed.

### Verifying MSTP Configuration on Switch 2

**Purpose** Verify the MSTP configuration on Switch 2.

**Action** Issue the operational mode commands **show spanning-tree interface** and **show spanning-tree bridge**:

```
user@switch2> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/14 | 128:513 | 128:513               | 32768.0019e2503d20      | 1000         | FWD   | DESG |
| ge-0/0/18 | 128:519 | 128:515               | 8192.0019e25051e0       | 1000         | FWD   | ROOT |

Spanning tree interface parameters for instance 1

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/14 | 128:513 | 128:513               | 32769.0019e2503d20      | 1000         | FWD   | DESG |
| ge-0/0/18 | 128:519 | 128:515               | 4097.0019e25051e0       | 1000         | FWD   | ROOT |

Spanning tree interface parameters for instance 2

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/14 | 128:513 | 128:513               | 4098.0019e2503d20       | 1000         | FWD   | DESG |
| ge-0/0/18 | 128:519 | 128:519               | 4098.0019e2503d20       | 1000         | FWD   | DESG |

```
user@switch2> show spanning-tree bridge
```

STP bridge parameters

```
Context ID : 0
Enabled protocol : MSTP
```

STP bridge parameters for CIST

```
Root ID : 8192.00:19:e2:50:51:e0
Root cost : 1000
Root port : ge-0/0/18
CIST regional root : 8192.00:19:e2:50:51:e0
CIST internal root cost : 1000
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Hop count : 19
Message age : 0
Number of topology changes : 1
Time since last topology change : 782 seconds
Local parameters
```

```

Bridge ID : 32768.00:19:e2:50:3d:20
Extended system ID : 0
Internal instance ID : 0

STP bridge parameters for MSTI 1
MSTI regional root : 4096.00:19:e2:50:51:e0
Root cost : 1000
Root port : ge-0/0/18
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Hop count : 19
Local parameters
 Bridge ID : 32768.00:19:e2:50:3d:20
 Extended system ID : 0
 Internal instance ID : 1

STP bridge parameters for MSTI 2
MSTI regional root : 4096.00:19:e2:50:3d:20
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Local parameters
 Bridge ID : 4096.00:19:e2:50:3d:20
 Extended system ID : 0
 Internal instance ID : 2

```

**Meaning** The operational mode command **show spanning-tree interface** displays spanning-tree domain information such as the designated port and the port roles. The spanning-tree interface parameters for instance 2 show that both ports are designated ports, which means Switch 2 is the root bridge for this instance.

The operational mode command **show spanning-tree bridge** displays the spanning-tree domain information at either the bridge level or interface level. If the optional interface name is omitted, all interfaces in the spanning-tree domain are displayed.

### Verifying MSTP Configuration on Switch 3

**Purpose** Verify the MSTP configuration on Switch 3.

**Action** Issue the operational mode commands **show spanning-tree interface** and **show spanning-tree bridge**:

```
user@switch3> show spanning-tree interface
```

```
Spanning tree interface parameters for instance 0
```

| Interface | Port ID | Designated port ID | Designated bridge ID | Port Cost | State | Role |
|-----------|---------|--------------------|----------------------|-----------|-------|------|
| ge-0/0/26 | 128:513 | 128:513            | 8192.0019e25051e0    | 1000      | FWD   | DESG |
| ge-0/0/28 | 128:515 | 128:515            | 8192.0019e25051e0    | 1000      | FWD   | DESG |
| ge-0/0/24 | 128:517 | 128:517            | 8192.0019e25051e0    | 1000      | FWD   | DESG |

```
Spanning tree interface parameters for instance 1
```

| Interface | Port ID | Designated port ID | Designated bridge ID | Port Cost | State | Role |
|-----------|---------|--------------------|----------------------|-----------|-------|------|
| ge-0/0/26 | 128:513 | 128:513            | 4096.0019e25051e0    | 1000      | FWD   | DESG |

|           |         |         |                   |      |     |      |
|-----------|---------|---------|-------------------|------|-----|------|
| ge-0/0/28 | 128:515 | 128:515 | 4096.0019e25051e0 | 1000 | FWD | DESG |
| ge-0/0/24 | 128:517 | 128:517 | 4096.0019e25051e0 | 1000 | FWD | DESG |

Spanning tree interface parameters for instance 2

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/26 | 128:513 | 128:531               | 8192.0019e25044e0       | 1000         | BLK   | ALT  |
| ge-0/0/28 | 128:515 | 128:519               | 4096.0019e2503d20       | 1000         | FWD   | ROOT |
| ge-0/0/24 | 128:517 | 128:517               | 16384.0019e25051e0      | 1000         | FWD   | DESG |

user@switch3> **show spanning-tree bridge**

STP bridge parameters

Context ID : 0  
Enabled protocol : MSTP

STP bridge parameters for CIST

Root ID : 8192.00:19:e2:50:51:e0  
CIST regional root : 8192.00:19:e2:50:51:e0  
CIST internal root cost : 0  
Hello time : 2 seconds  
Maximum age : 20 seconds  
Forward delay : 15 seconds  
Number of topology changes : 3  
Time since last topology change : 843 seconds  
Local parameters  
Bridge ID : 8192.00:19:e2:50:51:e0  
Extended system ID : 0  
Internal instance ID : 0

STP bridge parameters for MSTI 1

MSTI regional root : 4096.00:19:e2:50:51:e0  
Hello time : 2 seconds  
Maximum age : 20 seconds  
Forward delay : 15 seconds  
Local parameters  
Bridge ID : 4096.00:19:e2:50:51:e0  
Extended system ID : 0  
Internal instance ID : 1

STP bridge parameters for MSTI 2

MSTI regional root : 4096.00:19:e2:50:3d:20  
Root cost : 1000  
Root port : ge-0/0/28  
Hello time : 2 seconds  
Maximum age : 20 seconds  
Forward delay : 15 seconds  
Hop count : 19  
Local parameters  
Bridge ID : 16384.00:19:e2:50:51:e0  
Extended system ID : 0  
Internal instance ID : 2

**Meaning** The operational mode command **show spanning-tree interface** displays spanning-tree domain information such as the designated port and the port roles. Switch 3 is the root bridge for instance 0, which is the CIST, as well as for instance 1. In both instances, all ports on Switch 3 are designated ports.

The operational mode command **show spanning-tree bridge** displays the spanning-tree domain information at either the bridge level or the interface level. If the optional interface name is omitted, all interfaces in the spanning-tree domain are displayed.

### Verifying MSTP Configuration on Switch 4

**Purpose** Verify the MSTP configuration on Switch 4.

**Action** Issue the operational mode commands **show spanning-tree interface** and **show spanning-tree bridge**:

```
user@switch4> show spanning-tree interface
```

```
Spanning tree interface parameters for instance 0
```

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/23 | 128:523 | 128:517               | 8192.0019e25051e0       | 1000         | FWD   | ROOT |
| ge-0/0/19 | 128:525 | 128:525               | 16384.0019e25040e0      | 1000         | FWD   | DESG |

```
Spanning tree interface parameters for instance 1
```

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/23 | 128:523 | 128:517               | 4096.0019e25051e0       | 1000         | FWD   | ROOT |
| ge-0/0/19 | 128:525 | 128:525               | 16384.0019e25040e0      | 1000         | FWD   | DESG |

```
Spanning tree interface parameters for instance 2
```

| Interface | Port ID | Designated<br>port ID | Designated<br>bridge ID | Port<br>Cost | State | Role |
|-----------|---------|-----------------------|-------------------------|--------------|-------|------|
| ge-0/0/23 | 128:523 | 128:517               | 16384.0019e25051e0      | 1000         | BLK   | ALT  |
| ge-0/0/19 | 128:525 | 128:527               | 8192.0019e25044e0       | 1000         | FWD   | ROOT |

```
user@switch4> show spanning-tree bridge
```

```
STP bridge parameters
```

```
Context ID : 0
Enabled protocol : MSTP
```

```
STP bridge parameters for CIST
```

```
Root ID : 8192.00:19:e2:50:51:e0
Root cost : 0
Root port : ge-0/0/23
CIST regional root : 8192.00:19:e2:50:51:e0
CIST internal root cost : 1000
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Hop count : 19
Message age : 0
Number of topology changes : 4
Time since last topology change : 887 seconds
Local parameters
Bridge ID : 16384.00:19:e2:50:40:e0
Extended system ID : 0
Internal instance ID : 0
```

```

STP bridge parameters for MSTI 1
MSTI regional root : 4096.00:19:e2:50:51:e0
Root cost : 1000
Root port : ge-0/0/23
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Hop count : 19
Local parameters
 Bridge ID : 16384.00:19:e2:50:40:e0
 Extended system ID : 0
 Internal instance ID : 1

STP bridge parameters for MSTI 2
MSTI regional root : 4096.00:19:e2:50:3d:20
Root cost : 2000
Root port : ge-0/0/19
Hello time : 2 seconds
Maximum age : 20 seconds
Forward delay : 15 seconds
Hop count : 18
Local parameters
 Bridge ID : 32768.00:19:e2:50:40:e0
 Extended system ID : 0
 Internal instance ID : 2

```

**Meaning** The operational mode command **show spanning-tree interface** displays spanning-tree domain information such as the designated port and the port roles.

The operational mode command **show spanning-tree bridge** displays the spanning-tree domain information at either the bridge level or the interface level. If the optional interface name is omitted, all interfaces in the spanning-tree domain are displayed.

- Related Documentation**
- [Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123](#)
  - *Understanding MSTP for EX Series Switches*

## Understanding RSTP

Juniper Networks QFX Series products use Rapid Spanning Tree Protocol (RSTP) on the network side of the QFX Series to provide quicker convergence time than the base Spanning Tree Protocol (STP) does. RSTP identifies certain links as point to point. When a point-to-point link fails, the alternate link can transition to the forwarding state, which speeds up convergence.

Although STP provides basic loop prevention functionality, it does not provide fast network convergence when there are topology changes. The STP process to determine network state transitions is slower than the RSTP process because it is timer-based. A device must reinitialize every time a topology change occurs. The device must start in the listening state and transition to the learning state and eventually to a forwarding or blocking state. When default values are used for the maximum age (20 seconds) and forward delay (15 seconds), it takes 50 seconds for the device to converge. RSTP converges faster

because it uses a handshake mechanism based on point-to-point links instead of the timer-based process used by STP.

For networks with virtual LANs (VLANs), you can use VLAN Spanning Tree Protocol (VSTP), which takes the paths of each VLAN into account when calculating routes. VSTP uses RSTP by default.

An RSTP domain running from the edge outward on a QFX Series product has the following components:

- A *root port*, which is the “best path” to the root device.
- A *designated port*, which indicates that the switch is the designated bridge for the other switch connecting to this port.
- An *alternate port*, which provides an alternate root port.
- A *backup port*, which provides an alternate designated port.

Port assignments change through messages exchanged throughout the domain. An RSTP device generates configuration messages once per hello time interval. If an RSTP device does not receive a configuration message from its neighbor after an interval of three hello times, it determines that the connection with the neighbor is lost. When a *root port* or a *designated port* fails on a device, the device generates a configuration message with the proposal bit set. Once its neighbor device receives this message, it verifies that this configuration message is valid for that port and starts a *synchronizing* operation to ensure that all of its ports are in sync with the new information.

Similar sets of messages propagate through the network, restoring the connectivity very quickly after a topology change (in a well-designed network that uses RSTP, network convergence can take as little as 0.5 seconds). If a device does not receive an agreement to a proposal message it has sent, it returns to the original IEEE 802.D convention.

RSTP was originally defined in the IEEE 802.1w draft specification and later incorporated into the IEEE 802.1D-2004 specification.

VSTP and RSTP can be configured at the same time. If you configure VSTP and RSTP at the same time and the switch has more than 253 VLANs, VSTP is configured only for the first 253 VLANs. For the remaining VLANs, only RSTP is configured. RSTP and VSTP are the only spanning-tree protocols that can be configured at the same time on the QFX Series.



**NOTE:** Using the same VLAN for RSTP and VSTP is not supported. For example, if you are configuring a VLAN under VSTP, configuring RSTP with an interface that contains the same VLAN is not supported.

---

**Related Documentation**

- [Overview of Spanning-Tree Protocols on page 100](#)
- [Understanding MSTP on page 101](#)
- [Understanding VSTP on page 140](#)



- *Example: Configuring Faster Convergence and Improving Network Stability with RSTP*

## Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches



**NOTE:** This example uses Junos OS for EX Series switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Example: Faster Convergence and Improved Network Stability with RSTP on EX Series Switches*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

EX Series switches use Rapid Spanning Tree Protocol (RSTP) by default to provide a loop-free topology.

When switches that support redundant Routing Engines use RSTP, it is important to keep RSTP synchronized on both Routing Engines so that no loss of service occurs after a Routing Engine switchover. Nonstop bridging protocol keeps Routing Engines synchronized.

This example describes how to configure RSTP and NSB on four EX Series switches:

- [Requirements on page 123](#)
- [Overview and Topology on page 124](#)
- [Configuring RSTP and Nonstop Bridging on Switch 1 on page 126](#)
- [Configuring RSTP and Nonstop Bridging on Switch 2 on page 129](#)
- [Configuring RSTP and Nonstop Bridging on Switch 3 on page 131](#)
- [Configuring RSTP and Nonstop Bridging on Switch 4 on page 135](#)
- [Verification on page 137](#)

### Requirements

This example uses the following hardware and software components:

- Junos OS Release 13.2X50-D10 or later or later for EX Series switches
- Four EX Series switches

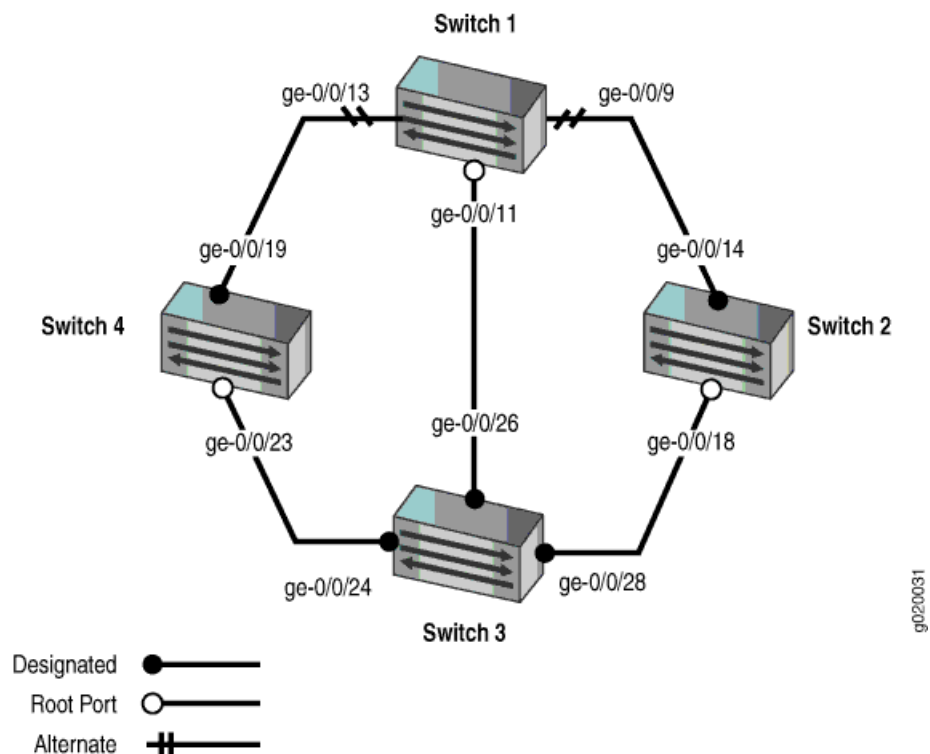
Before you configure the switches for RSTP, be sure you have:

- Installed and connected the four switches. See the hardware documentation for your switch.
- Performed the initial software configuration on all switches. See *Connecting and Configuring an EX Series Switch (CLI Procedure)* or *Connecting and Configuring an EX Series Switch (J-Web Procedure)*.

## Overview and Topology

RSTP works by identifying certain links as point to point links and blocking other possible paths. When one of the point-to-point links fails, a designated alternate link transitions to the forwarding state and take over. Configuring nonstop bridging (NSB) on a switch with redundant Routing Engines keeps RSTP synchronized on both Routing Engines. This way, RSTP remains active immediately after a switchover because it is already synchronized to the backup Routing Engine. RSTP does not have to reconverge after a Routing Engine switchover when NSB is enabled because the neighbor devices do not detect an RSTP change on the switch. In this example, four EX Series switches are connected in the topology displayed in [Figure 6 on page 124](#) to create a loop-free topology with NSB applied to switches with dual Routing Engines.

**Figure 6: Network Topology for RSTP**



[Table 11 on page 125](#) shows the components of the topology for this example.



**NOTE:** You can configure RSTP only on physical interfaces, not on logical interfaces.

Table 11: Components of the Topology for Configuring RSTP

| Property               | Settings                                                                                                                                                                                                                                                                       |
|------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Switch 1               | <p>The following interfaces on Switch 1 are connected in this way:</p> <ul style="list-style-type: none"> <li>• <b>ge-0/0/9</b> is connected to Switch 2</li> <li>• <b>ge-0/0/13</b> is connected to Switch 4</li> <li>• <b>ge-0/0/11</b> is connected to Switch 3</li> </ul>  |
| Switch 2               | <p>The following interfaces on Switch 2 are connected in this way:</p> <ul style="list-style-type: none"> <li>• <b>ge-0/0/14</b> is connected to Switch 1</li> <li>• <b>ge-0/0/18</b> is connected to Switch 3</li> </ul>                                                      |
| Switch 3               | <p>The following interfaces on Switch 3 are connected in this way:</p> <ul style="list-style-type: none"> <li>• <b>ge-0/0/26</b> is connected to Switch 1</li> <li>• <b>ge-0/0/28</b> is connected to Switch 2</li> <li>• <b>ge-0/0/24</b> is connected to Switch 4</li> </ul> |
| Switch 4               | <p>The following interfaces on Switch 4 are connected in this way:</p> <ul style="list-style-type: none"> <li>• <b>ge-0/0/19</b> is connected to Switch 1</li> <li>• <b>ge-0/0/23</b> is connected to Switch 3</li> </ul>                                                      |
| VLAN names and tag IDs | <p><b>voice-vlan</b>, tag 10<br/> <b>employee-vlan</b>, tag 20<br/> <b>guest-vlan</b>, tag 30<br/> <b>camera-vlan</b>, tag 40</p>                                                                                                                                              |

This configuration example creates a loop-free topology between four EX Series switches using RSTP.

An RSTP topology contains ports that have specific roles:

- The *root port* is responsible for forwarding data to the root bridge.
- The *alternate port* is a standby port for the root port. When a root port goes down, the alternate port becomes the active root port.
- The *designated port* forwards data to the downstream network segment or device.
- The *backup port* is a backup port for the designated port. When a designated port goes down, the backup port becomes the active designated port and starts forwarding data.



**NOTE:** You also can create a loop-free topology between the aggregation layer and the distribution layer using redundant trunk links. For more information about configuring redundant trunk links, see *Example: Configuring Redundant Trunk Links for Faster Recovery*.

## Configuring RSTP and Nonstop Bridging on Switch 1

|                                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CLI Quick Configuration</b> | <p>To quickly configure RSTP and nonstop bridging on Switch 1, copy the following commands and paste them into the switch terminal window:</p> <pre>[edit] set vlans voice-vlan description "Voice VLAN" set vlans voice-vlan vlan-id 10 set vlans employee-vlan description "Employee VLAN" set vlans employee-vlan vlan-id 20 set vlans guest-vlan description "Guest VLAN" set vlans guest-vlan vlan-id 30 set vlans camera-vlan description "Camera VLAN" set vlans camera-vlan vlan-id 40 set interfaces ge-0/0/13 unit 0 family ethernet-switching vlan members [10 20 30 40] set interfaces ge-0/0/9 unit 0 family ethernet-switching vlan members [10 20 30 40] set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members [10 20 30 40] set interfaces ge-0/0/13 unit 0 family ethernet-switching interface-mode trunk set interfaces ge-0/0/9 unit 0 family ethernet-switching interface-mode trunk set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk set protocols rstp bridge-priority 16k set protocols rstp interface ge-0/0/13 cost 1000 set protocols rstp interface ge-0/0/13 mode point-to-point set protocols rstp interface ge-0/0/9 cost 1000 set protocols rstp interface ge-0/0/9 mode point-to-point set protocols rstp interface ge-0/0/11 cost 1000 set protocols rstp interface ge-0/0/11 mode point-to-point</pre> <p>If Switch 1 includes dual Routing Engines, configure NSB. To quickly configure nonstop bridging on Switch 1, copy the following commands and paste them into the switch terminal window:</p> <pre>set chassis redundancy graceful switchover set system commit synchronize set protocols layer2-control nonstop-bridging</pre> |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step-by-Step Procedure</b> | <p>To configure RSTP and nonstop bridging on Switch 1:</p> <ol style="list-style-type: none"> <li>Configure the VLANs <b>voice-vlan</b>, <b>employee-vlan</b>, <b>guest-vlan</b>, and <b>camera-vlan</b>: <pre>[edit vlans] user@switch1# set voice-vlan description "Voice VLAN" user@switch1# set voice-vlan vlan-id 10 user@switch1# set employee-vlan description "Employee VLAN" user@switch1# set employee-vlan vlan-id 20 user@switch1# set guest-vlan description "Guest VLAN" user@switch1# set guest-vlan vlan-id 30 user@switch1# set camera-vlan description "Camera VLAN" user@switch1# set camera-vlan vlan-id 40</pre> </li> <li>Configure the VLANs on the interfaces, including support for the Ethernet switching protocol: <pre>[edit interfaces] user@switch1# set ge-0/0/13 unit 0 family ethernet-switching vlan members [10 20 30 40] user@switch1# set ge-0/0/9 unit 0 family ethernet-switching vlan members [10 20 30 40] user@switch1# set ge-0/0/11 unit 0 family ethernet-switching vlan members [10 20 30 40]</pre> </li> <li>Configure the port mode for the interfaces:</li> </ol> |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

- ```
[edit interfaces]
user@switch1# set ge-0/0/13 unit 0 family ethernet-switching interface-mode trunk
user@switch1# set ge-0/0/9 unit 0 family ethernet-switching interface-mode trunk
user@switch1# set ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk
```
4. Configure RSTP on the switch:

```
[edit protocols]
user@switch1# rstp bridge-priority 16k
user@switch1# rstp interface ge-0/0/13 cost 1000
user@switch1# rstp interface ge-0/0/13 mode point-to-point
user@switch1# rstp interface ge-0/0/9 cost 1000
user@switch1# rstp interface ge-0/0/9 mode point-to-point
user@switch1# rstp interface ge-0/0/11 cost 1000
user@switch1# rstp interface ge-0/0/11 mode point-to-point
```

Step-by-Step Procedure If Switch 1 includes dual Routing Engines, configure nonstop bridging. To configure NSB on Switch 1:

1. Enable graceful Routing Engine switchover (GRES):

```
[edit chassis redundancy]
user@switch1# set graceful-switchover
```

2. Configure the switch to always synchronize configuration changes between the Routing Engines:

```
[edit system]
user@switch1# set commit synchronize
```

If you try to commit a configuration in which nonstop bridging is configured but synchronization of configuration changes is not configured, the configuration is not committed.

3. Enable nonstop bridging:

```
[edit protocols layer2-control]
user@switch1# set nonstop-bridging
```



NOTE: This process enables NSB for all NSB-supported Layer 2 protocols on the switch, including RSTP.

Results Check the results of the configuration:

```
user@switch1> show configuration
interfaces {
  ge-0/0/13 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
  ge-0/0/9 {
```

```
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
  ge-0/0/11 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
}
protocols {
  layer2-control {
    nonstop-bridging;
  }
  rstp {
    bridge-priority 16k;
    interface ge-0/0/13 {
      cost 1000;
      mode point-to-point;
    }
    interface ge-0/0/9 {
      cost 1000;
      mode point-to-point;
    }
    interface ge-0/0/11 {
      cost 1000;
      mode point-to-point;
    }
  }
}
}
vlangs {
  voice-vlan {
    vlan-id 10;
  }
  employee-vlan {
    vlan-id 20;
  }
  guest-vlan {
    vlan-id 30;
  }
  camera-vlan {
    vlan-id 40;
  }
}
system {
```

```

    commit synchronize;
}
chassis {
    redundancy {
        graceful-switchover;
    }
}

```

Configuring RSTP and Nonstop Bridging on Switch 2

CLI Quick Configuration To quickly configure RSTP and nonstop bridging on Switch 2, copy the following commands and paste them into the switch terminal window:

```

[edit]
set vlans voice-vlan description "Voice VLAN"
set vlans voice-vlan vlan-id 10
set vlans employee-vlan description "Employee VLAN"
set vlans employee-vlan vlan-id 20
set vlans guest-vlan description "Guest VLAN"
set vlans guest-vlan vlan-id 30
set vlans camera-vlan description "Camera VLAN"
set vlans camera-vlan vlan-id 40
set interfaces ge-0/0/14 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/18 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/18 unit 0 family ethernet-switching interface-mode trunk
set protocols rstp bridge-priority 32k
set protocols rstp interface ge-0/0/14 cost 1000
set protocols rstp interface ge-0/0/14 mode point-to-point
set protocols rstp interface ge-0/0/18 cost 1000
set protocols rstp interface ge-0/0/18 mode point-to-point

```

If Switch 2 includes dual Routing Engines, configure NSB. To quickly configure nonstop bridging on Switch 2, copy the following commands and paste them into the switch terminal window:

```

set chassis redundancy graceful switchover
set system commit synchronize
set protocols layer2-control nonstop-bridging

```

Step-by-Step Procedure To configure RSTP and nonstop bridging on Switch 2:

1. Configure the VLANs **voice-vlan**, **employee-vlan**, **guest-vlan**, and **camera-vlan**:

```

[edit vlans]
user@switch2# set voice-vlan description "Voice VLAN"
user@switch2# set voice-vlan vlan-id 10
user@switch2# set employee-vlan description "Employee VLAN"
user@switch2# set employee-vlan vlan-id 20
user@switch2# set guest-vlan description "Guest VLAN"
user@switch2# set guest-vlan vlan-id 30
user@switch2# set camera-vlan vlan-description "Camera VLAN"
user@switch2# set camera-vlan vlan-id 40

```

2. Configure the VLANs on the interfaces, including support for the Ethernet switching protocol:

```

[edit interfaces]
user@switch2# set ge-0/0/14 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch2# set ge-0/0/18 unit 0 family ethernet-switching vlan members [10 20 30 40]

```

3. Configure the port mode for the interfaces:

```
[edit interfaces]
user@switch2# set ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk
user@switch2# set ge-0/0/18 unit 0 family ethernet-switching interface-mode trunk
```

4. Configure RSTP on the switch:

```
[edit protocols]
user@switch2# rstp bridge-priority 32k
user@switch2# rstp interface ge-0/0/14 cost 1000
user@switch2# rstp interface ge-0/0/14 mode point-to-point
user@switch2# rstp interface ge-0/0/18 cost 1000
user@switch2# rstp interface ge-0/0/18 mode point-to-point
```

Step-by-Step Procedure

If Switch 2 includes dual Routing Engines, configure nonstop bridging. To configure NSB on Switch 2:

1. Enable graceful Routing Engine switchover (GRES):

```
[edit chassis redundancy]
user@switch2# set graceful-switchover
```

2. Configure the switch to always synchronize configuration changes between the Routing Engines:

```
[edit system]
user@switch2# set commit synchronize
```

If you try to commit a configuration in which nonstop bridging is configured but synchronization of configuration changes is not configured, the configuration is not committed.

3. Enable nonstop bridging:

```
[edit protocols layer2-control]
user@switch2# set nonstop-bridging
```



NOTE: This process enables NSB for all NSB-supported Layer 2 protocols on the switch, including RSTP.

Results Check the results of the configuration:

```
user@switch2> show configuration
interfaces {
  ge-0/0/14 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
  ge-0/0/18 {
    unit 0 {
```



```

        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members [10 20 30 40];
            }
        }
    }
}
protocols {
    layer2-control {
        nonstop-bridging;
    }
    rstp {
        bridge-priority 32k;
        interface ge-0/0/14 {
            cost 1000;
            mode point-to-point;
        }
        interface ge-0/0/18 {
            cost 1000;
            mode point-to-point;
        }
    }
}
vpls {
    voice-vlan {
        vlan-id 10;
    }
    employee-vlan {
        vlan-id 20;
    }
    guest-vlan {
        vlan-id 30;
    }
    camera-vlan {
        vlan-id 40;
    }
}
system {
    commit synchronize;
}
chassis {
    redundancy {
        graceful-switchover;
    }
}

```

Configuring RSTP and Nonstop Bridging on Switch 3

CLI Quick Configuration To quickly configure RSTP and nonstop bridging on Switch 3, copy the following commands and paste them into the switch terminal window:

```

[edit]
set vlans voice-vlan description "Voice VLAN"
set vlans voice-vlan vlan-id 10

```

```

set vlans employee-vlan description "Employee VLAN"
set vlans employee-vlan vlan-id 20
set vlans guest-vlan description "Guest VLAN"
set vlans guest-vlan vlan-id 30
set vlans camera-vlan description "Camera VLAN"
set vlans camera-vlan vlan-id 40
set interfaces ge-0/0/26 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/28 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/24 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/26 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/28 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/24 unit 0 family ethernet-switching interface-mode trunk
set protocols rstp bridge-priority 8k
set protocols rstp interface ge-0/0/26 cost 1000
set protocols rstp interface ge-0/0/26 mode point-to-point
set protocols rstp interface ge-0/0/28 cost 1000
set protocols rstp interface ge-0/0/28 mode point-to-point
set protocols rstp interface ge-0/0/24 cost 1000
set protocols rstp interface ge-0/0/24 mode point-to-point

```

If Switch 3 includes dual Routing Engines, configure NSB. To quickly configure nonstop bridging on Switch 3, copy the following commands and paste them into the switch terminal window:

```

set chassis redundancy graceful switchover
set system commit synchronize
set protocols layer2-control nonstop-bridging

```

Step-by-Step Procedure

To configure RSTP and nonstop bridging on Switch 3:

1. Configure the VLANs **voice-vlan**, **employee-vlan**, **guest-vlan**, and **camera-vlan**:

```

[edit vlans]
user@switch3# set voice-vlan description "Voice VLAN"
user@switch3# set voice-vlan vlan-id 10
user@switch3# set employee-vlan description "Employee VLAN"
user@switch3# set employee-vlan vlan-id 20
user@switch3# set guest-vlan description "Guest VLAN"
user@switch3# set guest-vlan vlan-id 30
user@switch3# set camera-vlan description "Camera VLAN"
user@switch3# set camera-vlan vlan-id 40

```

2. Configure the VLANs on the interfaces, including support for the Ethernet switching protocol:

```

[edit interfaces]
user@switch3# set ge-0/0/26 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch3# set ge-0/0/28 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch3# set ge-0/0/24 unit 0 family ethernet-switching vlan members [10 20 30 40]

```

3. Configure the port mode for the interfaces:

```

[edit interfaces]
user@switch3# set ge-0/0/26 unit 0 family ethernet-switching interface-mode trunk
user@switch3# set ge-0/0/28 unit 0 family ethernet-switching interface-mode trunk
user@switch3# set ge-0/0/24 unit 0 family ethernet-switching interface-mode trunk

```

4. Configure RSTP on the switch:

```

[edit protocols]
user@switch3# rstp bridge-priority 8k

```

```

user@switch3# rstp interface ge-0/0/26 cost 1000
user@switch3# rstp interface ge-0/0/26 mode point-to-point
user@switch3# rstp interface ge-0/0/28 cost 1000
user@switch3# rstp interface ge-0/0/28 mode point-to-point
user@switch3# rstp interface ge-0/0/24 cost 1000
user@switch3# rstp interface ge-0/0/24 mode point-to-point

```

Step-by-Step Procedure If Switch 3 includes dual Routing Engines, configure nonstop bridging. To configure NSB on Switch 3:

1. Enable graceful Routing Engine switchover (GRES):

```

[edit chassis redundancy]
user@switch3# set graceful-switchover

```

2. Configure the switch to always synchronize configuration changes between the Routing Engines:

```

[edit system]
user@switch3# set commit synchronize

```

If you try to commit a configuration in which nonstop bridging is configured but synchronization of configuration changes is not configured, the configuration is not committed.

3. Enable nonstop bridging:

```

[edit protocols layer2-control]
user@switch3# set nonstop-bridging

```



NOTE: This process enables NSB for all NSB-supported Layer 2 protocols on the switch, including RSTP.

Results Check the results of the configuration:

```

user@switch3> show configuration
interfaces {
  ge-0/0/26 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
  ge-0/0/28 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
}

```

```
    }
    ge-0/0/24 {
      unit 0 {
        family ethernet-switching {
          interface-mode trunk;
          vlan {
            members [10 20 30 40];
          }
        }
      }
    }
  }
}
protocols {
  layer2-control {
    nonstop-bridging;
  }
  rstp {
    bridge-priority 8k;
    interface ge-0/0/26 {
      cost 1000;
      mode point-to-point;
    }
    interface ge-0/0/28 {
      cost 1000;
      mode point-to-point;
    }
    interface ge-0/0/24 {
      cost 1000;
      mode point-to-point;
    }
  }
  bridge-priority 8k;
}
}
}
vlangs {
  voice-vlan {
    vlan-id 10;
  }
  employee-vlan {
    vlan-id 20;
  }
  guest-vlan {
    vlan-id 30;
  }
  camera-vlan {
    vlan-id 40;
  }
}
system {
  commit synchronize;
}
chassis {
  redundancy {
```

```
    graceful-switchover;
}
```

Configuring RSTP and Nonstop Bridging on Switch 4

CLI Quick Configuration To quickly configure RSTP and nonstop bridging on Switch 4, copy the following commands and paste them into the switch terminal window:

```
[edit]
set vlans voice-vlan description "Voice VLAN"
set vlans voice-vlan vlan-id 10
set vlans employee-vlan description "Employee VLAN"
set vlans employee-vlan vlan-id 20
set vlans guest-vlan description "Guest VLAN"
set vlans guest-vlan vlan-id 30
set vlans camera-vlan description "Camera VLAN"
set vlans camera-vlan vlan-id 40
set interfaces ge-0/0/23 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/19 unit 0 family ethernet-switching vlan members [10 20 30 40]
set interfaces ge-0/0/23 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/19 unit 0 family ethernet-switching interface-mode trunk
set protocols rstp bridge-priority 16k
set protocols rstp interface ge-0/0/23 cost 1000
set protocols rstp interface ge-0/0/23 mode point-to-point
set protocols rstp interface ge-0/0/19 cost 1000
set protocols rstp interface ge-0/0/19 mode point-to-point
```

If Switch 4 includes dual Routing Engines, configure NSB. To quickly configure nonstop bridging on Switch 4, copy the following commands and paste them into the switch terminal window:

```
set chassis redundancy graceful switchover
set system commit synchronize
set protocols layer2-control nonstop-bridging
```

Step-by-Step Procedure To configure RSTP and nonstop bridging on Switch 4:

1. Configure the VLANs `voice-vlan`, `employee-vlan`, `guest-vlan`, and `camera-vlan`:

```
[edit vlans]
user@switch4# set voice-vlan description "Voice VLAN"
user@switch4# set voice-vlan vlan-id 10
user@switch4# set employee-vlan description "Employee VLAN"
user@switch4# set employee-vlan vlan-id 20
user@switch4# set guest-vlan description "Guest VLAN"
user@switch4# set guest-vlan vlan-id 30
user@switch4# set camera-vlan description "Camera VLAN"
user@switch4# set camera-vlan vlan-id 40
```

2. Configure the VLANs on the interfaces, including support for the Ethernet switching protocol:

```
[edit interfaces]
user@switch4# set ge-0/0/23 unit 0 family ethernet-switching vlan members [10 20 30 40]
user@switch4# set ge-0/0/19 unit 0 family ethernet-switching vlan members [10 20 30 40]
```

3. Configure the port mode for the interfaces:

```
[edit interfaces]
user@switch4# set ge-0/0/23 unit 0 family ethernet-switching interface-mode trunk
```

- user@switch4# set ge-0/0/19 unit 0 family ethernet-switching interface-mode trunk
4. Configure RSTP on the switch:

```
[edit protocols]
user@switch4# rstp bridge-priority 16k
user@switch4# rstp interface ge-0/0/23 cost 1000
user@switch4# rstp interface ge-0/0/23 mode point-to-point
user@switch4# rstp interface ge-0/0/19 cost 1000
user@switch4# rstp interface ge-0/0/19 mode point-to-point
```

Step-by-Step Procedure If Switch 4 includes dual Routing Engines, configure nonstop bridging. To configure NSB on Switch 4:

1. Enable graceful Routing Engine switchover (GRES):

```
[edit chassis redundancy]
user@switch4# set graceful-switchover
```

2. Configure the switch to always synchronize configuration changes between the Routing Engines:

```
[edit system]
user@switch4# set commit synchronize
```

If you try to commit a configuration in which nonstop bridging is configured but synchronization of configuration changes is not configured, the configuration is not committed.

3. Enable nonstop bridging:

```
[edit protocols layer2-control]
user@switch4# set nonstop-bridging
```



NOTE: This process enables NSB for all NSB-supported Layer 2 protocols on the switch, including RSTP.

Results Check the results of the configuration:

```
user@switch4> show configuration
interfaces {
  ge-0/0/23 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
  ge-0/0/19 {
    unit 0 {
      family ethernet-switching {
        interface-mode trunk;
        vlan {
          members [10 20 30 40];
        }
      }
    }
  }
}
```

```

    }
  }
}
}
protocols {
  layer2-control {
    nonstop-bridging;
  }
  rstp {
    bridge-priority 16k;
    interface ge-0/0/23 {
      cost 1000;
      mode point-to-point;
    }
    interface ge-0/0/19 {
      cost 1000;
      mode point-to-point;
    }
  }
}
}
vllans {
  voice-vlan {
    vlan-id 10;
  }
  employee-vlan {
    vlan-id 20;
  }
  guest-vlan {
    vlan-id 30;
  }
  camera-vlan {
    vlan-id 40;
  }
}
system {
  commit synchronize;
}
chassis {
  redundancy {
    graceful-switchover;
  }
}

```

Verification

To confirm that the configuration is working properly, perform these tasks on both Routing Engines:

- [Verifying RSTP Configuration on Switch 1 on page 138](#)
- [Verifying RSTP Configuration on Switch 2 on page 138](#)
- [Verifying RSTP Configuration on Switch 3 on page 138](#)
- [Verifying RSTP Configuration on Switch 4 on page 139](#)

Verifying RSTP Configuration on Switch 1

Purpose Verify the RSTP configuration on Switch 1.

Action Use the operational mode command:

```
user@switch1> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ge-0/0/13	128:526	128:526	16384.0019e25040e0	1000	BLK	ALT
ge-0/0/9	128:522	128:522	32768.0019e2503d20	1000	BLK	ALT
ge-0/0/11	128:524	128:524	8192.0019e25051e0	1000	FWD	ROOT

Meaning Refer to the topology in [Figure 6 on page 124](#). The operational mode command **show spanning-tree interface** shows that **ge-0/0/13** is in a forwarding state. The other interfaces on Switch 1 are blocking.

Verifying RSTP Configuration on Switch 2

Purpose Use this procedure to verify the RSTP configuration on both Switch 2 Routing Engines.

Action Use the operational mode command:

```
user@switch2> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ge-0/0/14	128:527	128:527	32768.0019e2503d20	1000	FWD	DESG
ge-0/0/18	128:529	128:529	8192.0019e25051e0	1000	FWD	ROOT

Meaning Refer to the topology in [Figure 6 on page 124](#). The operational mode command **show spanning-tree interface** shows that **ge-0/0/18** is in a forwarding state and is the root port.

Verifying RSTP Configuration on Switch 3

Purpose Use this procedure to verify the RSTP configuration on both Switch 3 Routing Engines.

Action Use the operational mode commands:

```
user@switch3> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ge-0/0/26	128:539	128:539	8192.0019e25051e0	1000	FWD	DESG
ge-0/0/28	128:541	128:541	8192.0019e25051e0	1000	FWD	DESG
ge-0/0/24	128:537	128:537	8192.0019e25051e0	1000	FWD	DESG

Meaning Refer to the topology in [Figure 6 on page 124](#). The operational mode command **show spanning-tree interface** shows that no interface is the root interface.

Verifying RSTP Configuration on Switch 4

Purpose Use this procedure to verify the RSTP configuration on both Switch 4 Routing Engines.

Action Use the operational mode commands:

```
user@switch4> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ge-0/0/23	128:536	128:536	8192.0019e25051e0	1000	FWD	ROOT
ge-0/0/19	128:532	128:532	16384.0019e25040e0	1000	FWD	DESG

Meaning Refer to the topology in [Figure 6 on page 124](#). The operational mode command **show spanning-tree interface** shows that interface **ge-0/0/23** is the root interface and forwarding.

Related Documentation

- *Understanding RSTP for EX Series Switches*

Configuring RSTP (CLI Procedure)

The default spanning-tree protocol for EX Series switches is Rapid Spanning Tree Protocol (RSTP). RSTP provides faster convergence times than the original Spanning Tree Protocol (STP). Because RSTP is configured by default, you only need to use this procedure if another spanning-tree protocol has been configured. In that case, you can reconfigure RSTP.

To enable RSTP:

1. Disable the other configured spanning-tree protocol (MSTP):
 - To disable MSTP:

```
[edit protocols]
user@switch# set mstp disable
```

2. Configure RSTP

- To enable RSTP

```
[edit protocols]
user@switch# set rstp
```

- To enable RSTP on a specific interface:



NOTE: You can configure RSTP only on physical interfaces, not on logical interfaces.

```
[edit protocols]
user@switch# set rstp interface interface-name
```

Related Documentation

- [Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123](#)
- *show spanning-tree bridge*
- *show spanning-tree interface*
- *Understanding RSTP for EX Series Switches*

Understanding VSTP

VLAN Spanning Tree Protocol (VSTP) enables Juniper Networks switches to run one or more Spanning Tree Protocol (STP) or Rapid Spanning Tree Protocol (RSTP) instances for each VLAN on which VSTP is enabled. For networks with multiple VLANs, VSTP improves intelligent tree spanning by defining best paths within the VLANs instead of within the entire network.

You can configure VSTP for a maximum of 509 VLANs.

VSTP and RSTP can be configured at the same time. If you configure VSTP and RSTP at the same time and the switch has more than 253 VLANs, VSTP is configured only for the first 253 VLANs. For the remaining VLANs, only RSTP is configured. RSTP and VSTP are the only spanning-tree protocols that can be configured at the same time on a switch.



NOTE: We recommend that you enable VSTP on all VLANs that could receive VSTP bridge protocol data units (BPDUs).



NOTE: Using the same VLAN for RSTP and VSTP is not supported. For example, if you are configuring a VLAN under VSTP, configuring RSTP with an interface that contains the same VLAN is not supported.

- Related Documentation**
- [Example: Configuring VSTP on QFX Series Switches and EX4600 Switches on page 141](#)
 - [Overview of Spanning-Tree Protocols on page 100](#)
 - [Understanding RSTP on page 121](#)
 - *Configuring VLAN Spanning Tree Protocol*
 - *Configuring VLAN Spanning Tree Protocol*
 - *vstp*

Example: Configuring VSTP on QFX Series Switches and EX4600 Switches

This example demonstrates configuring VSTP (VLAN Spanning Tree Protocol) on QFX Series switches and EX4600 switches. The default spanning-tree protocol on these switches is Rapid Spanning Tree Protocol (RSTP). VLAN Spanning Tree Protocol (VSTP) is an alternate protocol that allows switches to run one or more Spanning Tree Protocol (STP) or RSTP instances for each VLAN on which VSTP is enabled. For networks with multiple VLANs, VSTP improves network bandwidth utilization by load balancing traffic for each VLAN.

- [Requirements on page 141](#)
- [Overview on page 141](#)
- [Configuration on page 143](#)

Requirements

This example uses the following hardware and software components:

- QFX5100 switch (Note that any QFX Series switch or the EX4600 switch can be substituted.)
- Junos OS Release 13.2X51-D25 or later for the QFX Series or Junos OS Release 13.2X51-D25 or later for EX Series switches

Before you configure VSTP, be sure you have:

- Configured interfaces ge-0/0/3, ge-0/0/4, ge-0/0/8, ae0. For directions, see *Configuring Gigabit and 10-Gigabit Ethernet Interfaces*.
- Configured VLAN 200 with four interfaces. For directions, see [“Configuring VLANs” on page 66](#).
- Configured VLAN 5 and VLAN 6 with three interfaces each. For directions, see [“Configuring VLANs” on page 66](#).

Overview

You can configure VSTP for an interface at the global level (for all configured VLANs) or for a specific VLAN.

**NOTE:**

- If you configure VSTP on an interface at both the global and the specific VLAN level, the interface configuration that is defined at the specific VLAN level overrides the interface configuration that is defined at the global level.
- If you specify VSTP to be configured on an interface that is not configured to belong to the VLAN (or VLANs), an error message is displayed.

For each VLAN configured with VSTP, a dedicated instance of a spanning tree is created. This approach is useful to optimize network usage in small networks with a limited number of VLANs.

In this example, you configure VSTP at the VLAN level for VLAN 200 and at the global level for VLAN 5 and VLAN 6.



NOTE: You can use Juniper Networks switches with VSTP and Cisco switches with PVST+ and Rapid-PVST+ in the same network. Cisco supports a proprietary Per-VLAN Spanning Tree (PVST) protocol, which maintains a separate spanning tree instance per each VLAN. One Spanning Tree per VLAN allows fine grain load balancing but requires more BPDU CPU processing as the number of VLANs increases. PVST runs on Cisco proprietary ISL trunks which is not supported by Juniper. Juniper switches only inter-operate with PVST+ and Rapid-PVST+.

Topology

Table 12: Interfaces of the Topology for Configuring VSTP

Interface	Description
ge-0/0/3	Promiscuous member port
ge-0/0/4	Promiscuous member port
ge-0/0/8	Promiscuous member port
ae0	Promiscuous member port
VLAN ID	Description
200	Primary VLAN
5	Primary VLAN
6	Primary VLAN

Configuration

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

```
set protocols vstp vlan 200 interface ge-0/0/3
set protocols vstp vlan 200 interface ge-0/0/4
set protocols vstp vlan 200 interface ge-0/0/8
set protocols vstp vlan 200 interface ae0
set protocols vstp interface ge-0/0/3
set protocols vstp interface ge-0/0/4
set protocols vstp interface ge-0/0/8
set protocols vstp vlan 5
set protocols vstp vlan 6
```

Configuring

Step-by-Step Procedure To configure VSTP:

1. Enable VSTP on VLAN 200 using a single VLAN ID:

```
[edit protocols]
user@switch #set vstp vlan 200 interface ge-0/0/3
user@switch #set vstp vlan 200 interface ge-0/0/4
user@switch #set vstp vlan 200 interface ge-0/0/8
user@switch #set vstp vlan 200 interface ae0
```



NOTE: When you configure VSTP with the `set protocol vstp vlan all` command, VLAN ID 1 is not set; it is excluded so that the configuration is compatible with Cisco PVST+. If you want VLAN ID 1 to be included in the VSTP configuration on your switch, you must set it separately with the `set protocol vstp vlan 1` command.



TIP: You could also enable VSTP on a VLAN using a single VLAN name.

2. Enable VSTP on VLAN 5 and VLAN 6 at the VSTP global level:

```
[edit protocols]
user@switch #set vstp interface ge-0/0/3
user@switch #set vstp interface ge-0/0/4
user@switch #set vstp interface ge-0/0/8
user@switch #set vstp vlan 5
user@switch #set vstp vlan 6
```



CAUTION: Ensure that the interface is a member of all VLANs before you add the interface to the VSTP configuration. If the interface is not

a member of all VLANs, this VSTP configuration will fail when you try to commit it.

Verifying VSTP Configuration

Purpose View the spanning tree interface and bridge you just created.

Action To view a VSTP configuration, use the commands `show spanning-tree bridge` and `show spanning-tree interface`.

Results

The results of this example are:

```
root@sw-16> show spanning-tree interface
```

Spanning tree interface parameters for VLAN 5

Interface Port	State	Role	Port ID	Designated port ID	Designated bridge ID
Cost					
ge-0/0/3			128:490	128:490	32773.88e0f31f4040
20000	FWD	DESG			
ge-0/0/4			128:491	128:490	32773.88e0f31f4040
20000	BLK	BKUP			
ge-0/0/8			128:492	128:492	32773.88e0f31f4040
20000	FWD	DESG			

Spanning tree interface parameters for VLAN 6

Interface Port	State	Role	Port ID	Designated port ID	Designated bridge ID
Cost					
ge-0/0/3			128:490	128:490	32774.88e0f31f4040
20000	FWD	DESG			
ge-0/0/4			128:491	128:491	32774.88e0f31f4040
20000	FWD	DESG			
ge-0/0/8			128:492	128:492	32774.88e0f31f4040
20000	FWD	DESG			

Spanning tree interface parameters for VLAN 200

Interface Port	State	Role	Port ID	Designated port ID	Designated bridge ID
Cost					
ge-0/0/3			128:490	128:490	32968.88e0f31f4040
20000	FWD	DESG			
ge-0/0/4			128:491	128:491	32968.88e0f31f4040
20000	FWD	DESG			
ge-0/0/8			128:492	128:492	32968.88e0f31f4040
20000	FWD	DESG			
ae0			128:3	128:3	32968.88e0f31f4040

20000 FWD DESG

```
{master:0}
root@sw-16> show spanning-tree bridge
STP bridge parameters
Routing instance name           : GLOBAL
Enabled protocol                : RSTP

STP bridge parameters for VLAN 5
Root ID                        : 32773.88:e0:f3:1f:40:40
Hello time                     : 2 seconds
Maximum age                    : 20 seconds
Forward delay                  : 15 seconds
Message age                    : 0
Number of topology changes     : 1
Time since last topology change : 52 seconds
Local parameters
  Bridge ID                    : 32773.88:e0:f3:1f:40:40
  Extended system ID           : 5

STP bridge parameters for VLAN 6
Root ID                        : 32774.88:e0:f3:1f:40:40
Hello time                     : 2 seconds
Maximum age                    : 20 seconds
Forward delay                  : 15 seconds
Message age                    : 0
Number of topology changes     : 0
Local parameters
  Bridge ID                    : 32774.88:e0:f3:1f:40:40
  Extended system ID           : 6

STP bridge parameters for VLAN 200
Root ID                        : 32968.88:e0:f3:1f:40:40
Hello time                     : 2 seconds
Maximum age                    : 20 seconds
Forward delay                  : 15 seconds
Message age                    : 0
Number of topology changes     : 0
Local parameters
  Bridge ID                    : 32968.88:e0:f3:1f:40:40
  Extended system ID           : 200

{master:0}
root@sw-16>
```

Related Documentation • *Understanding VSTP for EX Series Switches and QFX Series Switches*

Understanding BPDU Protection for STP, RSTP, and MSTP



NOTE: Using the original CLI, you can disable BPDU protection on interfaces by issuing the `set ethernet-switching-options bpdu-block interface-name disable` command.

A Juniper Networks device Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), VLAN Spanning Tree Protocol (VSTP), and Multiple Spanning Tree Protocol (MSTP). Bridge protocol data unit (BPDU) protection can help prevent STP misconfigurations that can lead to network outages.

A loop-free network is supported through the exchange of a special type of frame called a BPDU. Receipt of BPDUs on certain interfaces in an STP, RSTP, VSTP, or MSTP topology, however, can lead to network outages. Enable BPDU protection on those interfaces to prevent these outages.

Peer STP applications running on the device interfaces use BPDUs to communicate. Ultimately, the exchange of BPDUs determines which interfaces block traffic and which interfaces become root ports and forward traffic.

However, a user bridge application running on a device connected to the device can also generate BPDUs. If these BPDUs are picked up by STP applications running on the device, they can trigger STP miscalculations, and those miscalculations can lead to network outages.

Enable BPDU protection on device interfaces connected to user devices or on interfaces on which no BPDUs are expected, such as edge ports. If BPDUs are received on a protected interface, the interface is disabled and stops forwarding frames.

Not only can you configure BPDU protection on a device with a spanning tree, but also on a device without a spanning tree. This type of topology typically consists of a non-STP device connected to an STP device through a trunk interface.

To configure BPDU protection on a device with a spanning tree, include the **bpdu-block-on-edge** statement at the `[edit protocols (stp | mstp | rstp)]` hierarchy level. To configure BPDU protection on a device without a spanning tree, include the **bpdu-block** statement at the `[edit ethernet-switching-options interface interface-name]` hierarchy level.

If BPDUs are sent to an interface (indicating that the misconfiguration has been corrected), the interface can be unblocked in one of two ways:

- If the **disable-timeout** statement has been included in the BPDU configuration, the interface automatically returns to service after the timer expires.
- Use the `clear error bpdu interface` operational mode command.

Disabling the BPDU protection configuration does not unblock the interface.

- Related Documentation**
- [Example: Configuring BPDU Protection on STP Interfaces to Prevent STP Miscalculations](#)
 - [Understanding Loop Protection for STP, RSTP, VSTP, and MSTP on page 151](#)
 - [Understanding Root Protection for STP, RSTP, VSTP, and MSTP on page 156](#)
 - [Understanding MSTP on page 101](#)
 - [Understanding RSTP on page 121](#)
 - [Understanding VSTP on page 140](#)

Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations on EX Series Switches

EX Series switches provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP). All spanning-tree protocols use a special type of frame called a bridge protocol data unit (BPDU) to communicate. Other devices—PC bridging applications, for example, also use BPDUs and generate their own BPDUs. These different BPDUs are not compatible. When BPDUs generated by spanning-tree protocols are transmitted to a device that uses another type of BPDU, they can cause problems on the device. Similarly, if switches within a spanning-tree topology receive BPDUs from other devices, network outages can occur because of STP miscalculations.

This example configures BPDU protection on an EX Series switch that uses RSTP. The upstream configuration is done on the edge interfaces, where outside BPDUs are often received from other devices:

- [Requirements on page 147](#)
- [Overview and Topology on page 148](#)
- [Configuration on page 149](#)
- [Verification on page 149](#)

Requirements

This example uses the following hardware and software components:

- Two EX Series switches in an RSTP topology
- Junos OS Release 9.1 or later for EX Series switches

Before you configure the interfaces on Switch 2 for BPDU protection, be sure you have:

- RSTP enabled on the switches.



NOTE: By default, RSTP is enabled on all EX Series switches.

Overview and Topology

The switches, being in an RSTP topology, support a loop-free network through the exchange of BPDUs. Receipt of outside BPDUs in an STP, RSTP, or MSTP topology, however, can lead to network outages by triggering an STP misconfiguration. To prevent such outages, enable BPDU protection on STP interfaces that could receive outside BPDUs. If an outside BPDU is received on a BPDU-protected interface, the interface shuts down to prevent the outside BPDU from accessing the STP interface.

Figure 7 on page 148 shows the topology for this example. In this example, Switch 1 and Switch 2 are configured for RSTP and create a loop-free topology. The interfaces on Switch 2 are edge access ports—edge access ports frequently receive outside BPDUs generated by PC applications.

This example configures interface **ge-0/0/5** and interface **ge-0/0/6** as edge ports on Switch 2, and then configures BPDU protection on those ports. With BPDU protection enabled, these interfaces shut down when they encounter an outside BPDU sent by the PCs connected to Switch 2.

Figure 7: BPDU Protection Topology

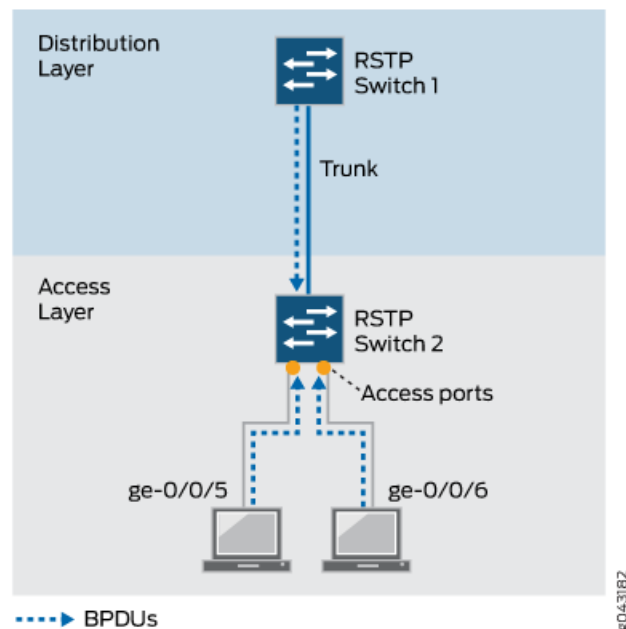


Table 13 on page 148 shows the components that will be configured for BPDU protection.

Table 13: Components of the Topology for Configuring BPDU Protection on EX Series Switches

Property	Settings
Switch 1 (Distribution Layer)	Switch 1 is connected to Switch 2 on a trunk interface.

Table 13: Components of the Topology for Configuring BPDU Protection on EX Series Switches (*continued*)

Property	Settings
Switch 2 (Access Layer)	Switch 2 has these access ports that require BPDU protection: <ul style="list-style-type: none"> • ge-0/0/5 • ge-0/0/6

This configuration example uses RSTP topology. You also can configure BPDU protection for STP or MSTP topologies at the `[edit protocols (mstp | stp)]` hierarchy level.

Configuration

To configure BPDU protection on two access interfaces:

CLI Quick Configuration Quickly configure RSTP on the two Switch 2 interfaces, and then configure BPDU protection on all edge ports on Switch 2 by copying the following commands and pasting them into the switch terminal window:

```
[edit]
set protocols rstp interface ge-0/0/5 edge
set protocols rstp interface ge-0/0/6 edge
set protocols rstp bpdu-block-on-edge
```

Step-by-Step Procedure To configure RSTP on the two Switch 2 interfaces, and then configure BPDU protection:

1. Configure RSTP on interface **ge-0/0/5** and interface **ge-0/0/6**, and configure them as edge ports:

```
[edit protocols rstp]
user@switch# set interface ge-0/0/5 edge
user@switch# set interface ge-0/0/6 edge
```

2. Configure BPDU protection on all edge ports on this switch:

```
[edit protocols rstp]
user@switch# set bpdu-block-on-edge
```

Results Check the results of the configuration:

```
user@switch> show configuration protocols rstp
interface ge-0/0/5 {
  edge;
}
interface ge-0/0/6 {
  edge;
}
bpdu-block-on-edge;
```

Verification

To confirm that the configuration is working properly:

- [Displaying the Interface State Before BPDU Protection Is Triggered on page 150](#)
- [Verifying That BPDU Protection Is Working Correctly on page 150](#)

Displaying the Interface State Before BPDU Protection Is Triggered

Purpose Before BPDUs can be received from PCs connected to interface **ge-0/0/5** and interface **ge-0/0/6**, confirm the interface state.

Action Use the operational mode command:

```
user@switch> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ge-0/0/0	128:513	128:513	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/1	128:514	128:514	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/2	128:515	128:515	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/3	128:516	128:516	32768.0019e2503f00	20000	FWD	DESG
ge-0/0/4	128:517	128:517	32768.0019e2503f00	20000	FWD	DESG
ge-0/0/5	128:518	128:518	32768.0019e2503f00	20000	FWD	DESG
ge-0/0/6	128:519	128:519	32768.0019e2503f00	20000	FWD	DESG

[output truncated]

Meaning The output from the operational mode command **show spanning-tree interface** shows that **ge-0/0/5** and interface **ge-0/0/6** are ports in a forwarding state.

Verifying That BPDU Protection Is Working Correctly

Purpose In this example, the PCs connected to Switch 2 start sending BPDUs to interface **ge-0/0/5** and interface **ge-0/0/6**. Verify that BPDU protection is working on the interfaces.

Action Use the operational mode command:

```
user@switch> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ge-0/0/0	128:513	128:513	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/1	128:514	128:514	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/2	128:515	128:515	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/3	128:516	128:516	32768.0019e2503f00	20000	FWD	DESG
ge-0/0/4	128:517	128:517	32768.0019e2503f00	20000	FWD	DESG
ge-0/0/5 (Bpdu-Incon)	128:518	128:518	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/6 (Bpdu-Incon)	128:519	128:519	32768.0019e2503f00	20000	BLK	DIS
ge-0/0/7	128:520	128:1	16384.00aabbcc0348	20000	FWD	ROOT
ge-0/0/8	128:521	128:521	32768.0019e2503f00	20000	FWD	DESG

[output truncated]

Meaning When BPDUs are sent from the PCs to interface **ge-0/0/5** and interface **ge-0/0/6** on Switch 2, the output from the operational mode command **show spanning-tree interface**

shows that the interfaces have transitioned to a BPDU inconsistent state. The BPDU inconsistent state causes the interfaces to shut down.

Disabling the BPDU protection configuration on an interface does not automatically reenable the interface. However, if the **disable-timeout (Spanning Trees)** statement has been included in the BPDU configuration, the interface does return to service after the timer expires. Otherwise, you must use the operational mode command **clear error bpdu** to unblock and reenable the interface.

If the PCs connected to Switch 2 send BPDUs to the interfaces again, BPDU protection is triggered once more and the interfaces transition back to the BPDU inconsistent state, causing them to shut down. In such cases, you need to find and repair the misconfiguration on the PCs that is sending BPDUs to Switch 2.

**Related
Documentation**

- *Example: Faster Convergence and Improved Network Stability with RSTP on EX Series Switches*
- *Example: Configuring BPDU Protection on Interfaces to Prevent STP Miscalculations on EX Series Switches*
- *Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on EX Series Switches*
- *Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees on EX Series Switches*
- *Understanding BPDU Protection for STP, RSTP, and MSTP on EX Series Switches*

Understanding Loop Protection for STP, RSTP, VSTP, and MSTP

A Juniper Networks device provides Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), VLAN Spanning Tree Protocol (VSTP), and Multiple Spanning Tree Protocol (MSTP). Loop protection increases the efficiency of STP, RSTP, and MSTP by preventing ports from entering a forwarding state that would cause a loop to open in the network.

A loop-free network in spanning-tree topologies is supported through the exchange of a special type of frame called a bridge protocol data unit (BPDU). Peer STP applications running on the device interfaces use BPDUs to communicate. Ultimately, the exchange of BPDUs determines which interfaces block traffic (preventing loops) and which interfaces become root ports and forward traffic.

However, a blocking interface can mistakenly transition to the forwarding state if the interface stops receiving BPDUs from its designated port on the segment. Such a transition error can occur when there is a hardware error on the device or software configuration error between the device and its neighbor.

When loop protection is enabled, the spanning-tree topology detects root ports and blocked ports and ensures that both keep receiving BPDUs. If a loop-protection-enabled interface stops receiving BPDUs from its designated port, it reacts as it would react to a problem with the physical connection on this interface. It does not transition the interface

to a forwarding state, but instead transitions it to a loop-inconsistent state. The interface recovers and it transitions back to the spanning-tree blocking state as soon as it receives a BPDU.

We recommend that you enable loop protection on all device interfaces that have a chance of becoming root or designated ports. Loop protection is most effective when enabled in the entire switched network. When you enable loop protection, you must configure at least one action (**alarm**, **block**, or both).

An interface can be configured for either loop protection or root protection, but not for both.

Related Documentation

- [Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on page 152](#)
- [Understanding Root Protection for STP, RSTP, VSTP, and MSTP on page 156](#)
- [Understanding BPDU Protection for STP, RSTP, and MSTP on page 146](#)
- [Understanding MSTP on page 101](#)
- [Understanding RSTP on page 121](#)
- [Overview of Spanning-Tree Protocols on page 100](#)
- [Understanding VSTP on page 140](#)

Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree

The QFX Series products provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP). Loop protection increases the efficiency of STP, RSTP, and MSTP by preventing interfaces from moving into a forwarding state that would create a loop in the network.

This example describes how to configure loop protection for an interface for the QFX Series in an RSTP topology:

- [Requirements on page 152](#)
- [Overview and Topology on page 153](#)
- [Configuration on page 154](#)
- [Verification on page 155](#)

Requirements

This example uses the following hardware and software components:

- Junos OS Release 11.1 or later for the QFX Series
- Three switches in an RSTP topology



NOTE: By default, RSTP is enabled for the QFX Series.

Overview and Topology

A loop-free network in spanning-tree topologies is supported through the exchange of a special type of frame called a bridge protocol data unit (BPDU). Peer STP applications running on the switch interfaces use BPDUs to communicate. Ultimately, the exchange of BPDUs determines which interfaces block traffic (preventing loops) and which interfaces become root ports and forward traffic.

A blocking interface can transition to the forwarding state in error if the interface stops receiving BPDUs from its designated port on the segment. Such a transition error can occur when there is a hardware error on the switch or software configuration error between the switch and its neighbor. When this happens, a loop appears in the spanning tree. Loops in a Layer 2 topology cause broadcast, unicast, and multicast frames to continuously circle the looped network. As a switch processes a flood of frames in a looped network, its resources become depleted, and the ultimate result is a network outage.



NOTE: An interface can be configured for either loop protection or root protection, but not for both.

Three switches are displayed in [Figure 8 on page 154](#). In this example, they are configured for RSTP and create a loop-free topology. Interface **xe-0/0/6** is blocking traffic between Switch 3 and Switch 1; thus, traffic is forwarded through interface **xe-0/0/7** on Switch 2. BPDUs are being sent from the root bridge on Switch 1 to both of these interfaces.

This example shows how to configure loop protection on interface **xe-0/0/6** to prevent it from transitioning from a blocking state to a forwarding state and creating a loop in the spanning-tree topology.

Figure 8: Network Topology for Loop Protection

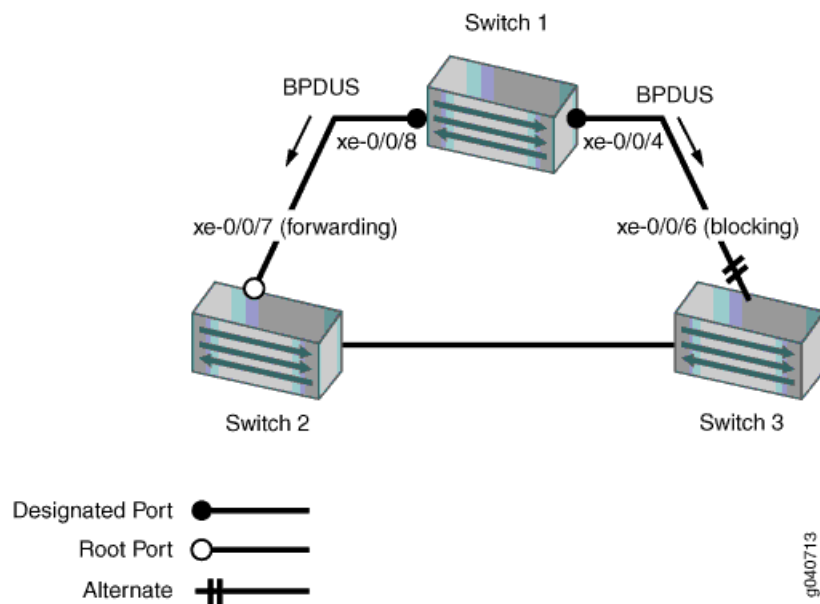


Table 14 on page 154 shows the components that will be configured for loop protection.

Table 14: Topology for Configuring Loop Protection on the QFX Series

Components	Settings
Switch 1	Switch 1 is the root bridge.
Switch 2	Switch 2 has the root port xe-0/0/7 .
Switch 3	Switch 3 is connected to Switch 1 through interface xe-0/0/6 .

A spanning-tree topology contains ports that have specific roles:

- The *root port* is responsible for forwarding data to the root bridge.
- The *alternate port* is a standby port for the root port. When a root port goes down, the alternate port becomes the active root port.
- The *designated port* forwards data to the downstream network segment or device.

This configuration example uses an RSTP topology. However, you can also configure loop protection for STP or MSTP topologies at the `[edit protocols (mstp | stp)]` hierarchy level.

Configuration

CLI Quick Configuration

To quickly configure loop protection on interface **xe-0/0/6**:

```
[edit]
set protocols rstp interface xe-0/0/6 bpdutimeout-action block
```


Step-by-Step Procedure To configure loop protection:

1. Configure interface **xe-0/0/6** on Switch 3:

```
[edit protocols rstp]
user@switch# set interface xe-0/0/6 bpdutimeout-action block
```

Results Check the results of the configuration:

```
user@switch> show configuration protocols rstp
interface xe-0/0/6.0 {
  bpdutimeout-action {
    block;
  }
}
```

Verification

To confirm that the configuration is working properly, perform these tasks:

- [Displaying the Interface State Before Loop Protection Is Triggered on page 155](#)
- [Verifying That Loop Protection Is Working on an Interface on page 155](#)

Displaying the Interface State Before Loop Protection Is Triggered

Purpose Before loop protection is triggered on interface **xe-0/0/6**, confirm that the interface is blocked.

Action Display the interface state and role before applying root protection:

```
user@switch> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
xe-0/0/0.0	128:513	128:513	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/1.0	128:514	128:514	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/2.0	128:515	128:515	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/3.0	128:516	128:516	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/4.0	128:517	128:517	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/5.0	128:518	128:518	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/6.0	128:519	128:2	16384.00aabbcc0348	20000	BLK	ALT

[output truncated]

Meaning The output from the operational mode command **show spanning-tree interface** shows that **xe-0/0/6.0** is the alternate port and is blocked.

Verifying That Loop Protection Is Working on an Interface

Purpose Verify that the loop protection configuration on interface **xe-0/0/6**. RSTP has been disabled on interface **xe-0/0/4** on Switch 1. This stops BPDUs from being sent to interface **xe-0/0/6** and triggering loop protection on that interface.

Action Display the interface state and role after applying root protection:

```
user@switch> show spanning-tree interface
```

```
Spanning tree interface parameters for instance 0
```

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
xe-0/0/0.0	128:513	128:513	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/1.0	128:514	128:514	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/2.0	128:515	128:515	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/3.0	128:516	128:516	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/4.0	128:517	128:517	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/5.0	128:518	128:518	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/6.0	128:519	128:519	32768.0019e2503f00	20000	BLK	DIS

(Loop-Incon)
[output truncated]

Meaning The operational mode command **show spanning-tree interface** shows that interface **xe-0/0/6.0** has detected that BPDUs are no longer being forwarded to it and has moved into a loop-inconsistent state. The loop-inconsistent state prevents the interface from transitioning to a forwarding state. The interface recovers and transitions back to its original state as soon as it receives BPDUs.

- Related Documentation**
- *Example: Configuring Faster Convergence and Improving Network Stability with RSTP*
 - [Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees on page 157](#)
 - *Example: Configuring BPDU Protection on STP Interfaces to Prevent STP Miscalculations*
 - [Understanding Loop Protection for STP, RSTP, VSTP, and MSTP on page 151](#)

Understanding Root Protection for STP, RSTP, VSTP, and MSTP

A Juniper Networks device provides Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), VLAN Spanning Tree Protocol (VSTP), and Multiple Spanning Tree Protocol (MSTP). A loop-free network is supported through the exchange of a special type of frame called a bridge protocol data unit (BPDU). Peer STP applications running on the device interfaces use BPDUs to communicate. Ultimately, the exchange of BPDUs determines which interfaces block traffic and which interfaces become root ports and forward traffic.

You can also see BPDUs generated when you run a bridge application on a device attached to the device. This can interfere with root port election, which may sometimes lead to the wrong root port being elected through the above process. Root protection allows you to manually enforce the root bridge placement in the network.

Enable root protection on interfaces that should not receive higher-priority BPDUs from the root bridge and should not be elected as the root port. These interfaces become designated ports and are typically located on an administrative boundary. If the bridge receives more STP BPDUs on a port that has root protection enabled, that port transitions

to a root-prevented STP state (inconsistency state), and the interface is blocked. This blocking prevents a bridge that should not be the root bridge from being elected the root bridge. After the bridge stops receiving more STP BPDUs on the interface with root protection, the interface returns to a listening state, followed by a learning state, and ultimately back to a forwarding state. Recovery back to the forwarding state is automatic.

When root protection is enabled on an interface, it is enabled for all the STP instances on that interface. The interface is blocked only for instances for which it receives more BPDUs. Otherwise, it participates in the spanning-tree topology.

An interface can be configured for either root protection or loop protection, but not for both.

Related Documentation

- [Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees on page 157](#)
- [Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on page 152](#)
- [Example: Configuring BPDU Protection on STP Interfaces to Prevent STP Miscalculations](#)
- [Understanding MSTP on page 101](#)
- [Understanding RSTP on page 121](#)
- [Overview of Spanning-Tree Protocols on page 100](#)
- [Understanding VSTP on page 140](#)

Example: Configuring Root Protection to Enforce Root Bridge Placement in Spanning Trees

QFX Series products provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP). Root protection increases the efficiency of STP, RSTP, and MSTP by allowing network administrators to enforce the root bridge placement in the network manually.

This example describes how to configure root protection on an interface for the QFX Series.

- [Requirements on page 157](#)
- [Overview and Topology on page 158](#)
- [Configuration on page 160](#)
- [Verification on page 160](#)

Requirements

This example uses the following hardware and software components:

- Junos OS Release 11.1 or later for the QFX Series
- Four switches in an RSTP topology

Before you configure the interface for root protection, be sure you have:

- RSTP operating on the switches.



NOTE: By default, RSTP is enabled on the QFX Series.

Overview and Topology

Peer STP applications running on switch interfaces exchange a special type of frame called a bridge protocol data unit (BPDU). Switches communicate interface information using BPDUs to create a loop-free topology that ultimately determines the root bridge and which interfaces block or forward traffic in the spanning tree.

You can also see BPDUs generated when you run a bridge application on a device attached to the switch. This can interfere with root port election, which may sometimes lead to the wrong root port being elected through the above process. Root protection allows you to manually enforce the root bridge placement in the network.

To prevent this from happening, enable root protection on interfaces that should not receive more BPDUs from the root bridge and should not be elected as the root port. These interfaces are typically located on an administrative boundary and are designated ports.

When root protection is enabled on an interface:

- The interface is blocked from becoming the root port.
- Root protection is enabled for all STP instances on that interface.
- The interface is blocked only for instances for which it receives more BPDUs. Otherwise, it participates in the spanning-tree topology.



NOTE: An interface can be configured for either root protection or loop protection, but not for both.

Four switches are displayed in [Figure 9 on page 159](#). In this example, they are configured for RSTP and create a loop-free topology. Interface **xe-0/0/7** on Switch 1 is a designated port on an administrative boundary. It connects to Switch 4. Switch 3 is the root bridge. Interface **xe-0/0/6** on Switch 1 is the root port.

This example shows how to configure root protection on interface **xe-0/0/7** to prevent it from transitioning to become the root port.

Figure 9: Network Topology for Root Protection

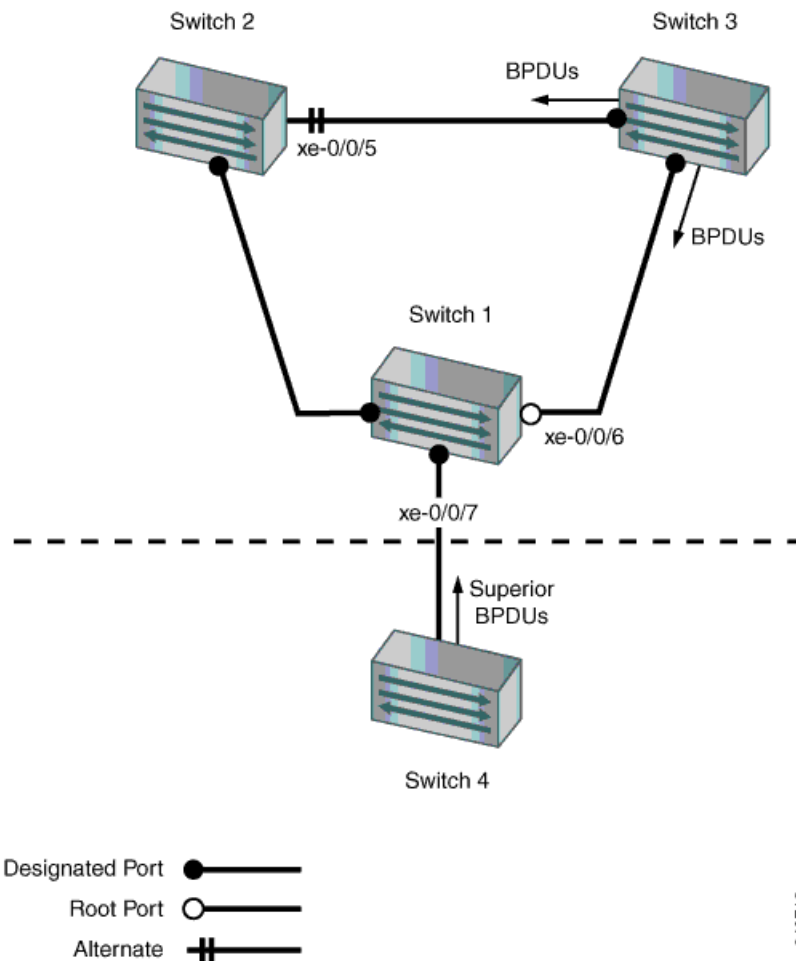


Table 15 on page 159 shows the components that will be configured for root protection.

Table 15: Topology for Configuring Root Protection on the QFX Series

Component	Settings
Switch 1	Switch 1 is connected to Switch 4 through interface xe-0/0/7 .
Switch 2	Switch 2 is connected to Switch 1 and Switch 3. Interface xe-0/0/4 is the alternate port in the RSTP topology.
Switch 3	Switch 3 is the root bridge and is connected to Switch 1 and Switch 2.
Switch 4	Switch 4 is connected to Switch 1. After loop protection is configured on interface xe-0/0/7 , Switch 4 sends more BPDUs that trigger loop protection on interface xe-0/0/7 .

A spanning-tree topology contains ports that have specific roles:

- The *root port* is responsible for forwarding data to the root bridge.

- The *alternate port* is a standby port for the root port. When a root port goes down, the alternate port becomes the active root port.
- The *designated port* forwards data to the downstream network segment or device.

This configuration example uses an RSTP topology. However, you can also configure root protection for STP or MSTP topologies at the `[edit protocols (mstp | stp)]` hierarchy level.

Configuration

CLI Quick Configuration	To quickly configure root protection on interface xe-0/0/7 , copy the following command and paste it into the switch terminal window: <pre>[edit] set protocols rstp interface xe-0/0/7 no-root-port</pre>
Step-by-Step Procedure	To configure root protection: 1. Configure interface xe-0/0/7 : <pre>[edit protocols rstp] user@switch# set interface xe-0/0/7 no-root-port</pre>
Results	Check the results of the configuration: <pre>user@switch> show configuration protocols rstp interface xe-0/0/7.0 { no-root-port; }</pre>

Verification

To confirm that the configuration is working properly, perform these tasks:

- [Displaying the Interface State Before Root Protection Is Triggered on page 160](#)
- [Verifying That Root Protection Is Working on the Interface on page 161](#)

Displaying the Interface State Before Root Protection Is Triggered

Purpose Before root protection is triggered on interface **xe-0/0/7**, confirm the interface state.

Action Confirm the state of the interfaces before root protection is configured:

```
user@switch> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
xe-0/0/0.0	128:513	128:513	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/1.0	128:514	128:514	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/2.0	128:515	128:515	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/3.0	128:516	128:516	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/4.0	128:517	128:517	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/5.0	128:518	128:2	16384.00aabbcc0348	20000	BLK	ALT
xe-0/0/6.0	128:519	128:1	16384.00aabbcc0348	20000	FWD	ROOT
xe-0/0/7.0	128:520	128:520	32768.0019e2503f00	20000	FWD	DESG

[output truncated]

Meaning The output from the operational mode command **show spanning-tree interface** shows that **xe-0/0/7.0** is a designated port in a forwarding state.

Verifying That Root Protection Is Working on the Interface

Purpose A configuration change takes place on Switch 4. A lower bridge priority on Switch 4 causes it to send more BPDUs to interface **xe-0/0/7**. Receipt of more BPDUs on interface **xe-0/0/7** triggers root protection. Verify that root protection is operating on interface **xe-0/0/7**.

Action Verify that root protection has been configured and is operating correctly:

```
user@switch> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
xe-0/0/0.0	128:513	128:513	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/1.0	128:514	128:514	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/2.0	128:515	128:515	32768.0019e2503f00	20000	BLK	DIS
xe-0/0/3.0	128:516	128:516	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/4.0	128:517	128:517	32768.0019e2503f00	20000	FWD	DESG
xe-0/0/5.0	128:518	128:2	16384.00aabbcc0348	20000	BLK	ALT
xe-0/0/6.0	128:519	128:1	16384.00aabbcc0348	20000	FWD	ROOT
xe-0/0/7.0	128:520	128:520	32768.0019e2503f00	20000	BLK	DIS

(Root-Incon)
[output truncated]

Meaning The operational mode command **show spanning-tree interface** shows that interface **xe-0/0/7.0** has transitioned to a loop inconsistent state. The loop inconsistent state blocks the interface and prevents it from becoming a candidate for the root port. When the root bridge no longer receives more STP BPDUs from the interface, the interface recovers and transitions back to a forwarding state. Recovery is automatic.

**Related
Documentation**

- *Example: Configuring Faster Convergence and Improving Network Stability with RSTP*
- [Example: Configuring Loop Protection to Prevent Interfaces from Transitioning from Blocking to Forwarding in a Spanning Tree on page 152](#)
- *Example: Configuring BPDU Protection on STP Interfaces to Prevent STP Miscalculations*
- [Understanding Root Protection for STP, RSTP, VSTP, and MSTP on page 156](#)

PART 5

Q-in-Q Tunneling

- [Using Q-in-Q Tunneling on page 165](#)

CHAPTER 5

Using Q-in-Q Tunneling

- [Understanding Q-in-Q Tunneling on page 165](#)
- [Configuring Q-in-Q Tunneling on page 170](#)
- [Configuring All-in-One Bundling on page 178](#)
- [Configuring Many-to-Many Bundling on page 180](#)
- [Configuring a Specific Interface Mapping with VLAN ID Translation Option on page 183](#)
- [Setting Up a Dual VLAN Tag Translation Configuration on page 185](#)

Understanding Q-in-Q Tunneling



NOTE: This topic applies to Junos OS switches with support for the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

Q-in-Q tunneling enables service providers on Ethernet access networks to extend a Layer 2 Ethernet connection between two customer sites. Using Q-in-Q tunneling, providers can also segregate or bundle customer traffic into fewer VLANs or different VLANs by adding another layer of 802.1Q tags. Q-in-Q tunneling is useful when customers have overlapping VLAN IDs because customers' VLAN (C-VLAN) tags are prepended by the service-provider VLAN (S-VLAN) tag, which allows you to preserve each customers' VLAN IDs without conflict. The Juniper Networks Junos operating system (Junos OS) implementation of Q-in-Q tunneling supports the IEEE 802.1ad standard.

This topic describes:

- [How Q-in-Q Tunneling Works on page 166](#)
- [How VLAN Translation Works on page 166](#)
- [Using Dual VLAN Tag Translation on page 167](#)
- [Sending and Receiving Untagged Packets on page 167](#)
- [Disabling MAC Address Learning on page 168](#)
- [Mapping C-VLANs to S-VLANs on page 168](#)
- [Constraints for Q-in-Q Tunneling and VLAN Translation on page 169](#)

How Q-in-Q Tunneling Works

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

When Q-in-Q tunneling is enabled, trunk interfaces are assumed to be part of the service provider or data center network. Access interfaces are assumed to be customer-facing and accept both tagged and untagged frames. This topic refers to trunk interfaces as S-VLAN interfaces. This type of interface is also known as a network-to-network interface (NNI). The topic refers to access interfaces as C-VLAN interfaces. This type of interface is also known as a user-network interface (UNI).



NOTE: Starting with Junos OS Release 14.1X53-D30, you can configure the same interface to be an S-VLAN/NNI interface and a C-VLAN/UNI interface, meaning that the same physical interface can transmit single-tagged and double-tagged frames simultaneously. This allows you maximum flexibility in your network topology and lets you maximize the use of your interfaces.

An interface can be a member of multiple S-VLANs. You can map one C-VLAN to one S-VLAN (1:1) or many C-VLANs to many S-VLANs (N:N). C-VLAN and S-VLAN tags are unique—for instance, you can have both a C-VLAN tag of 101 and an S-VLAN tag of 101. You can limit the set of accepted customer tags to a range of tags or to discrete values. Class-of-service (CoS) values of C-VLANs are unchanged in the downstream direction. You may copy ingress priority and CoS settings to the S-VLAN.

C-VLAN and S-VLAN interfaces accept priority-tagged packets without any configuration.

How VLAN Translation Works

VLAN translation replaces an incoming C-VLAN tag with an S-VLAN tag instead of adding an additional tag. The C-VLAN tag is therefore lost, so a single-tagged packet is normally untagged when it leaves the S-VLAN (at the other end of the link). If an incoming packet has had Q-in-Q tunneling applied in advance, VLAN translation replaces the outer tag and the inner tag is retained when the packet leaves the S-VLAN at the other end of the link.

To configure VLAN translation, use the *mapping swap* statement at the **[edit vlans interface]** hierarchy level.



NOTE: You can configure VLAN translation on access ports only. You cannot configure it on trunk ports, and you cannot configure Q-in-Q tunneling on the same access port.

Using Dual VLAN Tag Translation

Starting with Junos OS Release 14.1X53-D40, you can use the dual VLAN tag translation (also known as dual VLAN tag rewrite) feature to deploy switches in service-provider domains, allowing dual-tagged, single-tagged, and untagged VLAN packets to come into or exit from the switch. [Table 16 on page 167](#) shows the operations that are added for dual VLAN tag translation.

Table 16: Operations Added with Dual VLAN Tag Rewrite

Operation	Function
swap-push	Swap a VLAN tag and push a new VLAN tag
pop-swap	Pop an outer VLAN tag and swap an inner VLAN tag
swap-swap	Swap both outer and inner VLAN tags

Dual VLAN tag translation supports:

- Configuration of S-VLANs (NNI) and C-VLANs (UNI) on the same physical interface
- Control protocols such as VSTP, OSPF, and LACP
- IGMP snooping
- Configuration of a private VLAN (PVLAN) and VLAN on a single-tagged interface
- Use of TPID 0x8100 on both inner and outer VLAN tags

See [“Setting Up a Dual VLAN Tag Translation Configuration” on page 185](#).

Sending and Receiving Untagged Packets

To enable an interface to send and receive untagged packets, you must specify a native VLAN for a physical interface. When the interface receives an untagged packet, it adds the VLAN ID of the native VLAN to the packet and sends the newly tagged packet to the mapped interface.

To specify a native VLAN, use the **native-vlan-id** statement at the **[edit interfaces *interface-name*]** hierarchy level. The native VLAN ID must match the C-VLAN or S-VLAN ID or be included in the VLAN ID list specified on the logical interface.

For example, on a logical interface for a C-VLAN interface, you might specify a C-VLAN ID list of 100-200. Then, on the C-VLAN physical interface, you could specify a native VLAN ID of 150. This configuration would work because the native VLAN of 150 is included in the C-VLAN ID list of 100-200.

We recommend configuring a native VLAN when using any of the approaches to map C-VLANs to S-VLANs. If you do not configure a native VLAN on an interface, untagged packets received by the interface are discarded. See the Mapping C-VLANs to S-VLANs section in this topic for information about the methods of mapping C-VLANs to S-VLANs.

Disabling MAC Address Learning

In a Q-in-Q deployment, customer packets interfaces are transported without any changes to source and destination MAC addresses. You can disable MAC address learning at the global, interface, and VLAN levels:

- To disable learning globally, disable MAC address learning for the switch.
- To disable learning for an interface, disable MAC address learning for all VLANs of which the specified interface is a member.
- To disable learning for a VLAN, disable MAC address learning for a specified VLAN.

Mapping C-VLANs to S-VLANs

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

- [All-in-One Bundling on page 168](#)
- [Many-to-Many Bundling on page 168](#)
- [Mapping a Specific Interface on page 169](#)

If you configure multiple mapping methods, the switch gives priority to mapping a specific interface, then to many-to-many bundling, and last to all-in-one bundling. However, for a particular mapping method, setting up overlapping rules for the same C-VLAN is not supported.

All-in-One Bundling

All-in-one bundling maps all packets from all C-VLAN interfaces to an S-VLAN.

The C-VLAN interface accepts untagged and single-tagged packets. An S-VLAN 802.1Q tag is then added to these packets, and the packets are sent to the S-VLAN interface, which accepts untagged, single-tagged, and double-tagged packets.



NOTE: The C-VLAN and S-VLAN interfaces accept untagged packets provided that the `native-vlan-id` statement is configured on these interfaces.

Many-to-Many Bundling

Many-to-many bundling is used to specify which C-VLANs are mapped to which S-VLANs.

Use many-to-many bundling when you want a subset of the C-VLANs on the access switch to be part of multiple S-VLANs. With many-to-many bundling, the C-VLAN interfaces accept untagged and single-tagged packets. An S-VLAN 802.1Q tag is then added to these packets, and the packets are sent to the S-VLAN interfaces, which accept untagged, single-tagged, and double-tagged packets.



NOTE: The C-VLAN and S-VLAN interfaces accept untagged packets provided that the `native-vlan-id` statement is configured on these interfaces.

Mapping a Specific Interface

Use specific interface mapping when you want to assign an S-VLAN to a specific C-VLAN on an interface. The configuration applies only to the specific interface, not to all access interfaces.

Specific interface mapping has two suboptions: **push** and **swap**. When traffic that is mapped to a specific interface is pushed, the packet retains its original tag as it moves from the C-VLAN to the S-VLAN and an additional S-VLAN tag is added to the packet. When traffic that is mapped to a specific interface is swapped, the incoming tag is replaced with a new VLAN tag. This is sometimes known as VLAN rewriting or VLAN translation.

Typically, this method is used to keep data from different customers separate or to provide individualized treatment of the packets on a certain interface. You might also use this method to map VLAN traffic from different customers to a single S-VLAN.

When using specific interface mapping, the C-VLAN interfaces accept untagged and single-tagged packets, while the S-VLAN interfaces accept untagged, single-tagged, and double-tagged packets.



NOTE: The C-VLAN and S-VLAN interfaces accept untagged packets provided that the `native-vlan-id` statement is configured on these interfaces.

Constraints for Q-in-Q Tunneling and VLAN Translation

Be aware of the following constraints when configuring Q-in-Q tunneling and VLAN translation:

- With releases of Junos OS Release 13.2X51 previous to Release 13.2X51-D20, you cannot create a regular VLAN on an interface if you have created an S-VLAN or C-VLAN on that interface for Q-in-Q tunneling. This means that you cannot create an integrated routing and bridging (IRB) interface on that interface because regular VLANs are a required part of IRB configuration. With Junos OS Release 13.2X51-D25, you can create a regular VLAN on a trunk interface that has an S-VLAN, which means that you can also create an IRB interface on the trunk. In this case, the regular VLAN and S-VLAN on the same trunk interface cannot share the same VLAN ID. Junos OS Release 13.2X51-D25 does not allow you to create a regular VLAN on an access interface that has a C-VLAN.
- Starting with Junos OS Release 14.1X53-D40, integrated routing and bridging (IRB) interfaces are supported on Q-in-Q VLANs—you can configure the IRB interface on the same interface as one used by an S-VLAN, and you can use the same VLAN ID for both the VLAN used by the IRB interface and for the VLAN used as an S-VLAN.

Packets arriving on an IRB interface that is using Q-in-Q VLANs will get routed regardless of whether the packet is single tagged or double tagged. The outgoing routed packets contain an S-VLAN tag only when exiting a trunk interface; the packets exit the interface untagged when exiting an access interface.



NOTE: You can configure the IRB interface only on S-VLAN (NNI) interfaces, not on C-VLAN (UNI) interfaces.

- Most access port security features are not supported with Q-in-Q tunneling and VLAN translation.
- Configuring Q-in-Q tunneling and VLAN rewriting/VLAN translation on the same port is not supported.
- You can configure at most one VLAN rewrite/VLAN translation for a given VLAN and interface. For example, you can create no more than one translation for VLAN 100 on interface xe-0/0/0.
- The combined total of VLANs and rules for Q-in-Q tunneling and VLAN translation cannot exceed 6000. For example, you can configure and commit 4000 VLANs and 2000 rules for Q-in-Q tunneling and VLAN translation. However, you cannot configure 4000 VLANs and 2500 rules for Q-in-Q tunneling and VLAN translation. If you try to commit a configuration that exceeds the limit, you see CLI and syslog errors that inform you about the problem.
- MAC addresses are learned from S-VLANs, not C-VLANs.
- Broadcast, unknown unicast, and multicast traffic is forwarded to all members in the S-VLAN.
- The following features are not supported with Q-in-Q tunneling:
 - DHCP relay
 - Fibre Channel over Ethernet
 - IP Source Guard
- The following features are not supported with VLAN rewriting/VLAN translation:
 - Fibre Channel over Ethernet
 - Firewall filter applied to a port or VLAN in the output direction
 - Private VLANs
 - VLAN Spanning Tree Protocol
 - Reflective relay

**Related
Documentation**

- [Configuring Q-in-Q Tunneling on page 170](#)

Configuring Q-in-Q Tunneling

Q-in-Q tunneling and VLAN translation allow service providers to create a Layer 2 Ethernet connection between two customer sites. Providers can segregate different customers' VLAN traffic on a link (for example, if the customers use overlapping VLAN IDs) or bundle different customer VLANs into a single service VLAN. Data centers can use Q-in-Q

tunneling and VLAN translation to isolate customer traffic within a single site or to enable customer traffic flows between cloud data centers in different geographic locations.

Q-in-Q tunneling adds a service VLAN tag before the customer's 802.1Q VLAN tags. The Juniper Networks Junos operating system implementation of Q-in-Q tunneling supports the IEEE 802.1ad standard.



NOTE: This task uses a Junos OS release that supports the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Configuring Q-in-Q Tunneling*.

With releases of Junos OS 13.2X51 previous to 13.2X51-D20, you cannot create a regular VLAN on an interface if you have created an S-VLAN or C-VLAN on that interface for Q-in-Q tunneling. This means that you cannot create an integrated routing and bridging (IRB) interface on that interface because regular VLANs are a required part of IRB configuration. With Junos OS 13.2X51-D25, you can create a regular VLAN on a trunk interface that has an S-VLAN, which means that you can also create an IRB interface on the trunk. In this case, the regular VLAN and S-VLAN on the same trunk interface cannot share the same VLAN ID. Junos OS 13.2X51-D25 does not allow you to create a regular VLAN on an access interface that has a C-VLAN.

Before setting up Q-in-Q tunneling, make sure you have created and configured the necessary customer VLANs on the neighboring switches. See *Configuring VLANs*.

- [Using the Different Mapping Methods on page 171](#)
- [Configuring All-in-One Bundling on page 172](#)
- [Configuring Many-to-Many Bundling on page 173](#)
- [Configuring a Specific Interface Mapping with VLAN ID Translation Option on page 176](#)

Using the Different Mapping Methods

Once you have created the required VLANs on the neighboring switches, configure Q-in-Q tunneling using one of the three methods to map customer VLANs (C-VLANs) to service-provider-defined service VLANs (S-VLANs):

- All-in-one bundling maps all packets from all C-VLAN interfaces to an S-VLAN. For information about how to use this method, see [“Configuring All-in-One Bundling” on page 172](#).
- Use many-to-many bundling when you want a subset of the C-VLANs on the access switch to be part of multiple S-VLANs. For information about how to use this method, see [“Configuring Many-to-Many Bundling” on page 173](#).
- Use specific interface mapping when you want to assign an S-VLAN to a specific C-VLAN on an interface. For information about how to use this method, see [“Configuring a Specific Interface Mapping with VLAN ID Translation Option” on page 176](#).

Configuring All-in-One Bundling

You can configure Q-in-Q tunneling using the all-in-one bundling method, which forwards all packets that ingress on a C-VLAN interface to an S-VLAN. (Packets are forwarded to the S-VLAN regardless of whether they are tagged or untagged prior to ingress.) Using this approach saves you the effort of specifying a specific mapping for each C-VLAN.

First configure the S-VLAN and its interface:

1. Create the S-VLAN and assign a VLAN ID for it.

```
[edit vlans vlan-name]
user@switch# vlan-id vlan-id-number
```

2. Assign a logical interface (unit) to be a member of the S-VLAN.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```



NOTE: Do not use logical interface unit 0. You must later bind a VLAN tag ID to the unit you specify in this step, and you cannot bind a VLAN tag ID to unit 0.

3. Enable the interface to transmit packets with two 802.1Q VLAN tags:

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

4. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

5. Enable the S-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

6. Bind the logical interface (unit) of the interface that you specified in step 2 to the VLAN ID for the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id number
```

For example, the following configuration creates S-VLAN v10, makes xe-1/1/1.10 a member of that VLAN, enables Q-in-Q tunneling on interface xe-1/1/1, enables xe-1/1/1 to accept untagged packets, and binds the VLAN ID of S-VLAN v10 to a logical interface of xe-1/1/1.

```
set vlans v10 vlan-id 10
set vlans v10 interface xe-1/1/1.10
set interfaces xe-1/1/1 flexible-vlan-tagging
set interfaces xe-1/1/1 set native-vlan-id 10
set interfaces xe-1/1/1 encapsulation extended-vlan-bridge
set interfaces xe-1/1/1 unit 10 vlan-id 10
```

Now configure all-in-one bundling on a C-VLAN interface:

1. Assign a logical interface (unit) of the C-VLAN interface to be a member of the S-VLAN.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

2. Enable the interface to transmit packets with 802.1Q VLAN tags :

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

3. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

4. Enable the C-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

5. Configure a logical interface to receive and forward any tagged packet whose VLAN ID tag matches the list of VLAN IDs you specify:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id-list vlan-id-numbers
```



WARNING: On some EX and QFX Series switches, you can apply no more than eight VLAN identifier lists to a physical interface.

6. Configure the system to add an S-VLAN tag (outer tag) as packets travel from a C-VLAN interface to the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# input-vlan-map push
```

7. Configure the system to remove the S-VLAN tag when packets are forwarded (internally) from the S-VLAN interface to the C-VLAN interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# output-vlan-map pop
```

For example, the following configuration makes xe-0/0/1.10 a member of S-VLAN v10, enables Q-in-Q tunneling, maps packets from C-VLANs 100 through 200 to S-VLAN 10, and enables xe-0/0/1 to accept untagged packets. If a packet originates in C-VLAN 100 and needs to be sent across the S-VLAN, a tag with VLAN ID 10 is added to the packet. When a packet is forwarded (internally) from the S-VLAN interface to interface xe-0/0/1, the tag with VLAN ID 10 is removed.

```
set vlans v10 interface xe-0/0/1.10
set interfaces xe-0/0/1 flexible-vlan-tagging
set interfaces xe-0/0/1 encapsulation extended-vlan-bridge
set interfaces xe-0/0/1 unit 10 vlan-id-list 100-200
set interfaces xe-0/0/1 native-vlan-id 150
set interfaces xe-0/0/1 unit 10 input-vlan-map push
set interfaces xe-0/0/1 unit 10 output-vlan-map pop
```

Configuring Many-to-Many Bundling

You can configure Q-in-Q tunneling using the many-to-many bundling method, which maps packets from multiple C-VLANs to multiple S-VLANs. This method is convenient

for mapping a range of C-VLANs without having to specify each one individually. (You can also use this method to configure only one C-VLAN to be mapped to an S-VLAN.)

First configure the S-VLANs and assign them to an interface:

1. Create one of the S-VLANs and assign a VLAN ID for it.

```
[edit vlans vlan-name]  
user@switch# vlan-id vlan-id-number
```

2. Repeat step 1 for the other S-VLANs.

3. Assign a logical interface (unit) to be a member of one of the S-VLANs. Do not use logical interface unit 0.

```
[edit vlans vlan-name]  
user@switch# interface interface-name.unit-number
```

4. Repeat step 3 to assign the other logical interfaces on the same physical interface to be a member of other S-VLANs.

5. Enable the physical interface to transmit packets with two 802.1Q VLAN tags:

```
[edit interfaces interface-name]  
user@switch# flexible-vlan-tagging
```

6. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]  
user@switch# encapsulation extended-vlan-bridge
```

7. Enable the S-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]  
user@switch# set native-vlan-id vlan-id
```

8. Bind one of the logical units of the interface to the VLAN ID for one of the S-VLANs.

```
[edit interfaces interface-name unit logical-unit-number]  
user@switch# vlan-id number
```

9. Repeat step 8 to bind the VLAN IDs for the other S-VLANs to the other logical units of the interface:

For example, the following configuration creates S-VLANs v10 and v30 and associates them with interface xe-1/1/1. It also enables Q-in-Q tunneling, enables xe-1/1/1 to accept untagged packets, and maps incoming C-VLAN packets to S-VLANs v10 and v30.

```
set vlans v10 vlan-id 10  
set vlans v30 vlan-id 30  
set vlans v10 interface xe-1/1/1.10  
set vlans v30 interface xe-1/1/1.30  
set interfaces xe-1/1/1 flexible-vlan-tagging  
set interfaces xe-1/1/1 set native-vlan-id 10  
set interfaces xe-1/1/1 encapsulation extended-vlan-bridge  
set interfaces xe-1/1/1 unit 10 vlan-id 10  
set interfaces xe-1/1/1 unit 30 vlan-id 30
```

To configure the many-to-many bundling method on a C-VLAN interface, perform the following steps for each customer:

1. Assign a logical interface (unit) of one C-VLAN interface to be a member of one S-VLAN.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

2. Repeat step 1 to assign another C-VLAN interface (physical interface) to be a member of another S-VLAN.

3. Enable the interface to transmit packets with 802.1Q VLAN tags:

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

4. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

5. Enable the C-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

6. For each physical interface, configure a logical interface (unit) to receive and forward any tagged packet whose VLAN ID tag matches the list of VLAN IDs you specify:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id-list vlan-id-numbers
```



WARNING: On some EX and QFX Series switches, you can apply no more than eight VLAN identifier lists to a physical interface.

To configure only one C-VLAN to be mapped to an S-VLAN, specify only one VLAN ID after *vlan-id-list*.

7. For each physical interface, configure the system to add an S-VLAN tag (outer tag) as packets travel from the C-VLAN interface to the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# input-vlan-map push
```

8. For each physical interface, configure the system to remove the S-VLAN tag when packets are forwarded from the S-VLAN interface to the C-VLAN interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# output-vlan-map pop
```

For example, the following configuration makes xe-0/0/1.10 a member of S-VLAN v10, enables Q-in-Q tunneling, and maps packets from C-VLANs 10 through 20 to S-VLAN 10. The configuration for customer 2 makes xe-0/0/2.30 a member of S-VLAN v30, enables Q-in-Q tunneling, and maps packets from C-VLANs 30 through 40, 50 through 60, and 70 through 80 to S-VLAN 30. Both interfaces are configured to accept untagged packets.

If a packet originates in C-VLAN 10 and needs to be sent over the S-VLAN, a tag with a VLAN ID 10 is added to the packet. If a packet is forwarded internally from the S-VLAN

interface to xe-0/0/1.10, the tag with VLAN ID 10 is removed. The same principles apply to the C-VLANs configured on interface xe-0/0/2.



NOTE: Notice that you can use the same tag value for an S-VLAN and C-VLAN. For example, the configuration for customer 1 maps C-VLAN ID 10 to S-VLAN ID 10. C-VLAN and S-VLAN tags use separate name spaces, so this configuration is allowed.

Configuration for customer 1:

```
set vlans v10 interface xe-0/0/1.10
set interfaces xe-0/0/1 flexible-vlan-tagging
set interfaces xe-0/0/1 encapsulation extended-vlan-bridge
set interfaces xe-0/0/1 unit 10 vlan-id-list 10-20
set interfaces xe-0/0/1 native-vlan-id 15
set interfaces xe-0/0/1 unit 10 input-vlan-map push
set interfaces xe-0/0/1 unit 10 output-vlan-map pop
```

Configuration for customer 2:

```
set vlans v30 interface xe-0/0/2.30
set interfaces xe-0/0/2 flexible-vlan-tagging
set interfaces xe-0/0/2 encapsulation extended-vlan-bridge
set interfaces xe-0/0/2 unit 30 vlan-id-list 30-40
set interfaces xe-0/0/2 unit 30 vlan-id-list 50-60
set interfaces xe-0/0/2 unit 30 vlan-id-list 70-80
set interfaces xe-0/0/2 native-vlan-id 75
set interfaces xe-0/0/2 unit 30 input-vlan-map push
set interfaces xe-0/0/2 unit 30 output-vlan-map pop
```

Configuring a Specific Interface Mapping with VLAN ID Translation Option

You can configure Q-in-Q tunneling by mapping packets from a specified C-VLAN to a specified S-VLAN. In addition, you can configure the system to replace a C-VLAN tag with an S-VLAN tag or replace an S-VLAN tag with a C-VLAN tag (instead of double tagging). This is called VLAN translation or VLAN rewriting. VLAN translation is particularly useful if a service provider's Layer 2 network that connects a customer's sites does not support double tagged packets.

When you use VLAN translation, both ends of the link normally must be able to swap the tags appropriately. That is, both ends of the link must be configured to swap the C-VLAN tag for the S-VLAN tag and swap the S-VLAN tag for the C-VLAN tag so that traffic in both directions is tagged appropriately while in transit and after arrival.

First configure the S-VLAN and its interface:

1. Create the S-VLAN and assign a VLAN ID for it.

```
[edit vlans vlan-name]
user@switch# vlan-id vlan-id-number
```

2. Assign a logical interface to be a member of the S-VLAN. Do not use unit 0.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

3. Enable the interface to transmit packets with 802.1Q VLAN tags:

```
[edit interfaces interface-name]
```

```
user@switch# flexible-vlan-tagging
```

4. Enable the S-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
```

```
user@switch# set native-vlan-id vlan-id
```

5. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
```

```
user@switch# encapsulation extended-vlan-bridge
```

6. Bind the logical interface (unit) of the interface that you specified earlier to the VLAN ID for the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
```

```
user@switch# vlan-id number
```

For example, the following configuration creates S-VLAN v200, makes xe-1/1/1.200 a member of that VLAN, enables Q-in-Q tunneling on interface xe-1/1/1, enables xe-1/1/1 to accept untagged packets, and binds a logical interface of xe-1/1/1 to the VLAN ID of VLAN v200.

```
set vlans v200 vlan-id 200
```

```
set vlans v200 interface xe-1/1/1.200
```

```
set interfaces xe-1/1/1 flexible-vlan-tagging
```

```
set interfaces xe-1/1/1 set native-vlan-id 10
```

```
set interfaces xe-1/1/1 encapsulation extended-vlan-bridge
```

```
set interfaces xe-1/1/1 unit 200 vlan-id 200
```

Now configure a specific interface mapping with optional VLAN ID translation on the C-VLAN interface:

1. Assign a logical interface of the C-VLAN interface to be a member of the S-VLAN.

```
[edit vlans vlan-name]
```

```
user@switch# interface interface-name.unit-number
```

2. Enable the interface to transmit packets with 802.1Q VLAN tags:

```
[edit interfaces interface-name]
```

```
user@switch# flexible-vlan-tagging
```

3. Enable the C-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
```

```
user@switch# set native-vlan-id vlan-id
```

4. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
```

```
user@switch# encapsulation extended-vlan-bridge
```

5. Configure a logical interface (unit) to receive and forward any tagged packet whose VLAN ID tag matches the VLAN IDs you specify:

```
[edit interfaces interface-name unit logical-unit-number]
```

```
user@switch# vlan-id number
```

6. Configure the system to remove the existing C-VLAN tag and replace it with the S-VLAN tag when packets ingress on the C-VLAN interface and are forwarded to the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
```

```
user@switch# input-vlan-map swap
```

7. Configure the system to remove the existing S-VLAN tag and replace it with the C-VLAN tag when packets are forwarded from the S-VLAN interface to the C-VLAN interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# output-vlan-map swap
```

8. To configure an S-VLAN and associate it with the appropriate C-VLAN interface:

```
[edit vlans vlan-name]
user@switch# interface interface-name
```

For example, the following configuration on C-VLAN interface xe-0/0/1 enables Q-in-Q tunneling, enables xe-0/0/1 to accept untagged packets, and maps incoming packets from C-VLAN 150 to logical interface 200, which is a member of S-VLAN 200. Also, when packets egress from C-VLAN interface xe-0/0/1 and travel to the S-VLAN interface, the C-VLAN tag of 150 is removed and replaced with the S-VLAN tag of 200. When packets travel from the S-VLAN interface to the C-VLAN interface, the S-VLAN tag of 200 is removed and replaced with the C-VLAN tag of 150.

```
set vlans v200 interface xe-0/0/1.200
set interfaces xe-0/0/1 flexible-vlan-tagging
set interfaces xe-0/0/1 set native-vlan-id 10
set interfaces xe-0/0/1 encapsulation extended-vlan-bridge
set interfaces xe-0/0/1 unit 200 vlan-id 150
set interfaces xe-0/0/1 unit 200 output-vlan-map swap
set interfaces xe-0/0/1 unit 200 input-vlan-map swap
```

Related Documentation • [Understanding Q-in-Q Tunneling on page 165](#)

Configuring All-in-One Bundling

You can configure Q-in-Q tunneling using the all-in-one bundling method, which forwards all packets that ingress on a C-VLAN interface to an S-VLAN. (Packets are forwarded to the S-VLAN regardless of whether they are tagged or untagged prior to ingress.) Using this approach saves you the effort of specifying a specific mapping for each C-VLAN.

First configure the S-VLAN and its interface:

1. Create the S-VLAN and assign a VLAN ID for it.

```
[edit vlans vlan-name]
user@switch# vlan-id vlan-id-number
```

2. Assign a logical interface (unit) to be a member of the S-VLAN.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```



NOTE: Do not use logical interface unit 0. You must later bind a VLAN tag ID to the unit you specify in this step, and you cannot bind a VLAN tag ID to unit 0.

3. Enable the interface to transmit packets with two 802.1Q VLAN tags:

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

4. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```


5. Enable the S-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

6. Bind the logical interface (unit) of the interface that you specified in step 2 to the VLAN ID for the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id number
```

For example, the following configuration creates S-VLAN v10, makes xe-1/1/1.10 a member of that VLAN, enables Q-in-Q tunneling on interface xe-1/1/1, enables xe-1/1/1 to accept untagged packets, and binds the VLAN ID of S-VLAN v10 to a logical interface of xe-1/1/1.

```
set vlans v10 vlan-id 10
set vlans v10 interface xe-1/1/1.10
set interfaces xe-1/1/1 flexible-vlan-tagging
set interfaces xe-1/1/1 set native-vlan-id 10
set interfaces xe-1/1/1 encapsulation extended-vlan-bridge
set interfaces xe-1/1/1 unit 10 vlan-id 10
```

Now configure all-in-one bundling on a C-VLAN interface:

1. Assign a logical interface (unit) of the C-VLAN interface to be a member of the S-VLAN.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

2. Enable the interface to transmit packets with 802.1Q VLAN tags :

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

3. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

4. Enable the C-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

5. Configure a logical interface to receive and forward any tagged packet whose VLAN ID tag matches the list of VLAN IDs you specify:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id-list vlan-id-numbers
```



WARNING: On some EX and QFX Series switches, you can apply no more than eight VLAN identifier lists to a physical interface.

6. Configure the system to add an S-VLAN tag (outer tag) as packets travel from a C-VLAN interface to the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# input-vlan-map push
```

7. Configure the system to remove the S-VLAN tag when packets are forwarded (internally) from the S-VLAN interface to the C-VLAN interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# output-vlan-map pop
```

For example, the following configuration makes xe-0/0/1.10 a member of S-VLAN v10, enables Q-in-Q tunneling, maps packets from C-VLANs 100 through 200 to S-VLAN 10, and enables xe-0/0/1 to accept untagged packets. If a packet originates in C-VLAN 100 and needs to be sent across the S-VLAN, a tag with VLAN ID 10 is added to the packet. When a packet is forwarded (internally) from the S-VLAN interface to interface xe-0/0/1, the tag with VLAN ID 10 is removed.

```
set vlans v10 interface xe-0/0/1.10
set interfaces xe-0/0/1 flexible-vlan-tagging
set interfaces xe-0/0/1 encapsulation extended-vlan-bridge
set interfaces xe-0/0/1 unit 10 vlan-id-list 100-200
set interfaces xe-0/0/1 native-vlan-id 150
set interfaces xe-0/0/1 unit 10 input-vlan-map push
set interfaces xe-0/0/1 unit 10 output-vlan-map pop
```

- Related Documentation**
- [Understanding Q-in-Q Tunneling on page 165](#)
 - [Configuring Many-to-Many Bundling on page 173](#)
 - [Configuring a Specific Interface Mapping with VLAN ID Translation Option on page 176](#)

Configuring Many-to-Many Bundling

You can configure Q-in-Q tunneling using the many-to-many bundling method, which maps packets from multiple C-VLANs to multiple S-VLANs. This method is convenient for mapping a range of C-VLANs without having to specify each one individually. (You can also use this method to configure only one C-VLAN to be mapped to an S-VLAN.)

First configure the S-VLANs and assign them to an interface:

1. Create one of the S-VLANs and assign a VLAN ID for it.

```
[edit vlans vlan-name]
user@switch# vlan-id vlan-id-number
```

2. Repeat step 1 for the other S-VLANs.

3. Assign a logical interface (unit) to be a member of one of the S-VLANs. Do not use logical interface unit 0.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

4. Repeat step 3 to assign the other logical interfaces on the same physical interface to be a member of other S-VLANs.

5. Enable the physical interface to transmit packets with two 802.1Q VLAN tags:

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

6. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

7. Enable the S-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

8. Bind one of the logical units of the interface to the VLAN ID for one of the S-VLANs.

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id number
```

- Repeat step 8 to bind the VLAN IDs for the other S-VLANs to the other logical units of the interface:

For example, the following configuration creates S-VLANs v10 and v30 and associates them with interface xe-1/1/1. It also enables Q-in-Q tunneling, enables xe-1/1/1 to accept untagged packets, and maps incoming C-VLAN packets to S-VLANs v10 and v30.

```
set vlans v10 vlan-id 10
set vlans v30 vlan-id 30
set vlans v10 interface xe-1/1/1.10
set vlans v30 interface xe-1/1/1.30
set interfaces xe-1/1/1 flexible-vlan-tagging
set interfaces xe-1/1/1 set native-vlan-id 10
set interfaces xe-1/1/1 encapsulation extended-vlan-bridge
set interfaces xe-1/1/1 unit 10 vlan-id 10
set interfaces xe-1/1/1 unit 30 vlan-id 30
```

To configure the many-to-many bundling method on a C-VLAN interface, perform the following steps for each customer:

- Assign a logical interface (unit) of one C-VLAN interface to be a member of one S-VLAN.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

- Repeat step 1 to assign another C-VLAN interface (physical interface) to be a member of another S-VLAN.
- Enable the interface to transmit packets with 802.1Q VLAN tags:

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

- Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

- Enable the C-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

- For each physical interface, configure a logical interface (unit) to receive and forward any tagged packet whose VLAN ID tag matches the list of VLAN IDs you specify:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id-list vlan-id-numbers
```



WARNING: On some EX and QFX Series switches, you can apply no more than eight VLAN identifier lists to a physical interface.

To configure only one C-VLAN to be mapped to an S-VLAN, specify only one VLAN ID after *vlan-id-list*.

- For each physical interface, configure the system to add an S-VLAN tag (outer tag) as packets travel from the C-VLAN interface to the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
```

```
user@switch# input-vlan-map push
```

8. For each physical interface, configure the system to remove the S-VLAN tag when packets are forwarded from the S-VLAN interface to the C-VLAN interface:

```
[edit interfaces interface-name unit logical-unit-number]
```

```
user@switch# output-vlan-map pop
```

For example, the following configuration makes xe-0/0/1.10 a member of S-VLAN v10, enables Q-in-Q tunneling, and maps packets from C-VLANs 10 through 20 to S-VLAN 10. The configuration for customer 2 makes xe-0/0/2.30 a member of S-VLAN v30, enables Q-in-Q tunneling, and maps packets from C-VLANs 30 through 40, 50 through 60, and 70 through 80 to S-VLAN 30. Both interfaces are configured to accept untagged packets.

If a packet originates in C-VLAN 10 and needs to be sent over the S-VLAN, a tag with a VLAN ID 10 is added to the packet. If a packet is forwarded internally from the S-VLAN interface to xe-0/0/1.10, the tag with VLAN ID 10 is removed. The same principles apply to the C-VLANs configured on interface xe-0/0/2.



NOTE: Notice that you can use the same tag value for an S-VLAN and C-VLAN. For example, the configuration for customer 1 maps C-VLAN ID 10 to S-VLAN ID 10. C-VLAN and S-VLAN tags use separate name spaces, so this configuration is allowed.

Configuration for customer 1:

```
set vlans v10 interface xe-0/0/1.10
set interfaces xe-0/0/1 flexible-vlan-tagging
set interfaces xe-0/0/1 encapsulation extended-vlan-bridge
set interfaces xe-0/0/1 unit 10 vlan-id-list 10-20
set interfaces xe-0/0/1 native-vlan-id 15
set interfaces xe-0/0/1 unit 10 input-vlan-map push
set interfaces xe-0/0/1 unit 10 output-vlan-map pop
```

Configuration for customer 2:

```
set vlans v30 interface xe-0/0/2.30
set interfaces xe-0/0/2 flexible-vlan-tagging
set interfaces xe-0/0/2 encapsulation extended-vlan-bridge
set interfaces xe-0/0/2 unit 30 vlan-id-list 30-40
set interfaces xe-0/0/2 unit 30 vlan-id-list 50-60
set interfaces xe-0/0/2 unit 30 vlan-id-list 70-80
set interfaces xe-0/0/2 native-vlan-id 75
set interfaces xe-0/0/2 unit 30 input-vlan-map push
set interfaces xe-0/0/2 unit 30 output-vlan-map pop
```

Related Documentation

- [Understanding Q-in-Q Tunneling on page 165](#)
- [Configuring All-in-One Bundling on page 172](#)
- [Configuring a Specific Interface Mapping with VLAN ID Translation Option on page 176](#)

Configuring a Specific Interface Mapping with VLAN ID Translation Option

You can configure Q-in-Q tunneling by mapping packets from a specified C-VLAN to a specified S-VLAN. In addition, you can configure the system to replace a C-VLAN tag with an S-VLAN tag or replace an S-VLAN tag with a C-VLAN tag (instead of double tagging). This is called VLAN translation or VLAN rewriting. VLAN translation is particularly useful if a service provider's Layer 2 network that connects a customer's sites does not support double tagged packets.

When you use VLAN translation, both ends of the link normally must be able to swap the tags appropriately. That is, both ends of the link must be configured to swap the C-VLAN tag for the S-VLAN tag and swap the S-VLAN tag for the C-VLAN tag so that traffic in both directions is tagged appropriately while in transit and after arrival.

First configure the S-VLAN and its interface:

1. Create the S-VLAN and assign a VLAN ID for it.

```
[edit vlans vlan-name]
user@switch# vlan-id vlan-id-number
```

2. Assign a logical interface to be a member of the S-VLAN. Do not use unit 0.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

3. Enable the interface to transmit packets with 802.1Q VLAN tags:

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

4. Enable the S-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

5. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

6. Bind the logical interface (unit) of the interface that you specified earlier to the VLAN ID for the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id number
```

For example, the following configuration creates S-VLAN v200, makes xe-1/1/1.200 a member of that VLAN, enables Q-in-Q tunneling on interface xe-1/1/1, enables xe-1/1/1 to accept untagged packets, and binds a logical interface of xe-1/1/1 to the VLAN ID of VLAN v200.

```
set vlans v200 vlan-id 200
set vlans v200 interface xe-1/1/1.200
set interfaces xe-1/1/1 flexible-vlan-tagging
set interfaces xe-1/1/1 set native-vlan-id 10
set interfaces xe-1/1/1 encapsulation extended-vlan-bridge
set interfaces xe-1/1/1 unit 200 vlan-id 200
```

Now configure a specific interface mapping with optional VLAN ID translation on the C-VLAN interface:

1. Assign a logical interface of the C-VLAN interface to be a member of the S-VLAN.

```
[edit vlans vlan-name]
user@switch# interface interface-name.unit-number
```

2. Enable the interface to transmit packets with 802.1Q VLAN tags:

```
[edit interfaces interface-name]
user@switch# flexible-vlan-tagging
```

3. Enable the C-VLAN interface to send and receive untagged packets:

```
[edit interfaces interface-name]
user@switch# set native-vlan-id vlan-id
```

4. Enable extended VLAN bridge encapsulation on the interface:

```
[edit interfaces interface-name]
user@switch# encapsulation extended-vlan-bridge
```

5. Configure a logical interface (unit) to receive and forward any tagged packet whose VLAN ID tag matches the VLAN IDs you specify:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# vlan-id number
```

6. Configure the system to remove the existing C-VLAN tag and replace it with the S-VLAN tag when packets ingress on the C-VLAN interface and are forwarded to the S-VLAN:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# input-vlan-map swap
```

7. Configure the system to remove the existing S-VLAN tag and replace it with the C-VLAN tag when packets are forwarded from the S-VLAN interface to the C-VLAN interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@switch# output-vlan-map swap
```

8. To configure an S-VLAN and associate it with the appropriate C-VLAN interface:

```
[edit vlans vlan-name]
user@switch# interface interface-name
```

For example, the following configuration on C-VLAN interface xe-0/0/1 enables Q-in-Q tunneling, enables xe-0/0/1 to accept untagged packets, and maps incoming packets from C-VLAN 150 to logical interface 200, which is a member of S-VLAN 200. Also, when packets egress from C-VLAN interface xe-0/0/1 and travel to the S-VLAN interface, the C-VLAN tag of 150 is removed and replaced with the S-VLAN tag of 200. When packets travel from the S-VLAN interface to the C-VLAN interface, the S-VLAN tag of 200 is removed and replaced with the C-VLAN tag of 150.

```
set vlans v200 interface xe-0/0/1.200
set interfaces xe-0/0/1 flexible-vlan-tagging
set interfaces xe-0/0/1 set native-vlan-id 10
set interfaces xe-0/0/1 encapsulation extended-vlan-bridge
set interfaces xe-0/0/1 unit 200 vlan-id 150
set interfaces xe-0/0/1 unit 200 output-vlan-map swap
set interfaces xe-0/0/1 unit 200 input-vlan-map swap
```

Related Documentation

- [Understanding Q-in-Q Tunneling on page 165](#)

- [Configuring All-in-One Bundling on page 172](#)
- [Configuring Many-to-Many Bundling on page 173](#)

Setting Up a Dual VLAN Tag Translation Configuration

You can use the dual VLAN tag translation (also known as dual VLAN tag rewrite) feature to deploy switches in service-provider domains, allowing dual-tagged, single-tagged, and untagged VLAN packets to come into or exit from the switch.

The following example configuration shows use of the swap-swap, pop-swap, and swap-push dual tag operations.

```
[edit]
set interfaces ge-0/0/1 unit 503 description UNI-3
set interfaces ge-0/0/1 unit 503 encapsulation vlan-bridge
set interfaces ge-0/0/1 unit 503 vlan-tags outer 503
set interfaces ge-0/0/1 unit 503 vlan-tags inner 504
set interfaces ge-0/0/1 unit 503 input-vlan-map swap-swap
set interfaces ge-0/0/1 unit 503 input-vlan-map vlan-id 500
set interfaces ge-0/0/1 unit 503 input-vlan-map inner-vlan-id 514
set interfaces ge-0/0/1 unit 503 output-vlan-map swap-swap
set interfaces ge-0/0/0 description NNI
set interfaces ge-0/0/0 flexible-vlan-tagging
set interfaces ge-0/0/0 encapsulation flexible-ethernet-services
set interfaces ge-0/0/0 unit 500 description "SVLAN500 port"
set interfaces ge-0/0/0 unit 500 encapsulation vlan-bridge
set interfaces ge-0/0/0 unit 500 vlan-id 500
set interfaces ge-0/0/0 unit 600 description "SVLAN600 port"
set interfaces ge-0/0/0 unit 600 encapsulation vlan-bridge
set interfaces ge-0/0/0 unit 600 vlan-id 600
set interfaces ge-0/0/0 unit 700 description "SVLAN700 port"
set interfaces ge-0/0/0 unit 700 encapsulation vlan-bridge
set interfaces ge-0/0/0 unit 700 vlan-id 700
set interfaces ge-0/0/0 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/0 unit 0 family ethernet-switching vlan members v1000
set interfaces ge-0/0/0 unit 1100 description UNI-SVLAN1100
set interfaces ge-0/0/0 unit 1100 encapsulation vlan-bridge
set interfaces ge-0/0/0 unit 1100 vlan-tags outer 1101
set interfaces ge-0/0/0 unit 1100 vlan-tags inner 1102
set interfaces ge-0/0/0 unit 1100 input-vlan-map swap-swap
set interfaces ge-0/0/0 unit 1100 input-vlan-map vlan-id 1100
set interfaces ge-0/0/0 unit 1100 input-vlan-map inner-vlan-id 2101
set interfaces ge-0/0/0 unit 1100 output-vlan-map swap-swap
set interfaces ge-0/0/0 unit 1200 description UNI-SVLAN1200
set interfaces ge-0/0/0 unit 1200 encapsulation vlan-bridge
set interfaces ge-0/0/0 unit 1200 vlan-id 1201
set interfaces ge-0/0/0 unit 1200 input-vlan-map swap-push
set interfaces ge-0/0/0 unit 1200 input-vlan-map inner-vlan-id 2200
set interfaces ge-0/0/0 unit 1200 output-vlan-map pop-swap
set interfaces ge-0/0/2 description UNI
set interfaces ge-0/0/2 flexible-vlan-tagging
set interfaces ge-0/0/2 encapsulation flexible-ethernet-services
set interfaces ge-0/0/2 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/2 unit 0 family ethernet-switching vlan members v1000
set interfaces ge-0/0/2 unit 603 description UNI-3
set interfaces ge-0/0/2 unit 603 encapsulation vlan-bridge
set interfaces ge-0/0/2 unit 603 vlan-tags outer 603
```

```

set interfaces ge-0/0/2 unit 603 vlan-tags inner 604
set interfaces ge-0/0/2 unit 603 input-vlan-map swap-swap
set interfaces ge-0/0/2 unit 603 input-vlan-map vlan-id 600
set interfaces ge-0/0/2 unit 603 input-vlan-map inner-vlan-id 614
set interfaces ge-0/0/2 unit 603 output-vlan-map swap-swap
set interfaces ge-0/0/3 description UNI
set interfaces ge-0/0/3 flexible-vlan-tagging
set interfaces ge-0/0/3 encapsulation flexible-ethernet-services
set interfaces ge-0/0/3 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/3 unit 0 family ethernet-switching vlan members v1000
set interfaces ge-0/0/3 unit 703 description UNI-3
set interfaces ge-0/0/3 unit 703 encapsulation vlan-bridge
set interfaces ge-0/0/3 unit 703 vlan-tags outer 703
set interfaces ge-0/0/3 unit 703 vlan-tags inner 704
set interfaces ge-0/0/3 unit 703 input-vlan-map swap-swap
set interfaces ge-0/0/3 unit 703 input-vlan-map vlan-id 700
set interfaces ge-0/0/3 unit 703 input-vlan-map inner-vlan-id 714
set interfaces ge-0/0/3 unit 703 output-vlan-map swap-swap
set interfaces ge-0/0/3 unit 701 encapsulation vlan-bridge
set interfaces ge-0/0/3 unit 701 vlan-id 701
set interfaces ge-0/0/3 unit 701 input-vlan-map swap-push
set interfaces ge-0/0/3 unit 701 input-vlan-map inner-vlan-id 780
set interfaces ge-0/0/3 unit 701 output-vlan-map pop-swap
set interfaces ge-0/0/3 unit 1100 description SVLAN1100-NNI
set interfaces ge-0/0/3 unit 1100 encapsulation vlan-bridge
set interfaces ge-0/0/3 unit 1100 vlan-id 1100
set interfaces ge-0/0/3 unit 1200 description SVLAN1200-NNI
set interfaces ge-0/0/3 unit 1200 encapsulation vlan-bridge
set interfaces ge-0/0/3 unit 1200 vlan-id 1200
set vlans SVLAN500 interface ge-0/0/0.500
set vlans SVLAN500 interface ge-0/0/1.503
set vlans SVLAN600 interface ge-0/0/0.600
set vlans SVLAN600 interface ge-0/0/2.603
set vlans SVLAN600 interface ge-0/0/3.701
set vlans SVLAN700 interface ge-0/0/0.700
set vlans SVLAN700 interface ge-0/0/3.703
set vlans v1000 vlan-id 1000
set vlans SVLAN1100 interface ge-0/0/0.1100
set vlans SVLAN1100 interface ge-0/0/3.1100
set vlans SVLAN1200 interface ge-0/0/3.1200
set vlans SVLAN1200 interface ge-0/0/0.1200

```

- Related Documentation**
- [Understanding Q-in-Q Tunneling on page 165](#)
 - [Configuring All-in-One Bundling on page 172](#)
 - [Configuring Many-to-Many Bundling on page 173](#)

PART 6

Private VLANs

- [Using Private VLANs on page 189](#)

CHAPTER 6

Using Private VLANs

- [Understanding Private VLANs on page 189](#)
- [Understanding PVLAN Traffic Flows Across Multiple Switches on page 194](#)
- [Using an IRB Interface in a Private VLAN on page 198](#)
- [Example: Configuring a Private VLAN Spanning Multiple Switches With an IRB Interface on page 199](#)
- [Understanding Egress Firewall Filters with PVLANS on page 212](#)
- [Creating a Private VLAN on a Single Switch \(CLI Procedure\) on page 214](#)
- [Creating a Private VLAN Spanning Multiple Switches \(CLI Procedure\) on page 216](#)
- [Example: Configuring a Private VLAN on a Single Switch on page 217](#)
- [Verifying That a Private VLAN Is Working on page 221](#)

Understanding Private VLANs

VLANs limit broadcasts to specified users. Private VLANs (PVLANS) take this concept a step further by splitting the broadcast domain into multiple isolated broadcast subdomains and essentially putting secondary VLANs inside a primary VLAN. PVLANS restrict traffic flows through their member switch ports (called “private ports”) so that these ports communicate only with a specified uplink trunk port or with specified ports within the same VLAN. The uplink trunk port is usually connected to a router, firewall, server, or provider network. Each PVLAN typically contains many private ports that communicate only with a single uplink, thereby preventing the ports from communicating with each other.

Just like regular VLANs, PVLANS are isolated on Layer 2 and require that a Layer 3 device or an IRB interface be used to route traffic. PVLANS are useful for restricting the flow of broadcast and unknown unicast traffic and for limiting the communication between known hosts. Service providers use PVLANS to keep their customers isolated from one another.

This topic explains the following concepts regarding PVLANS on the QFX Series:

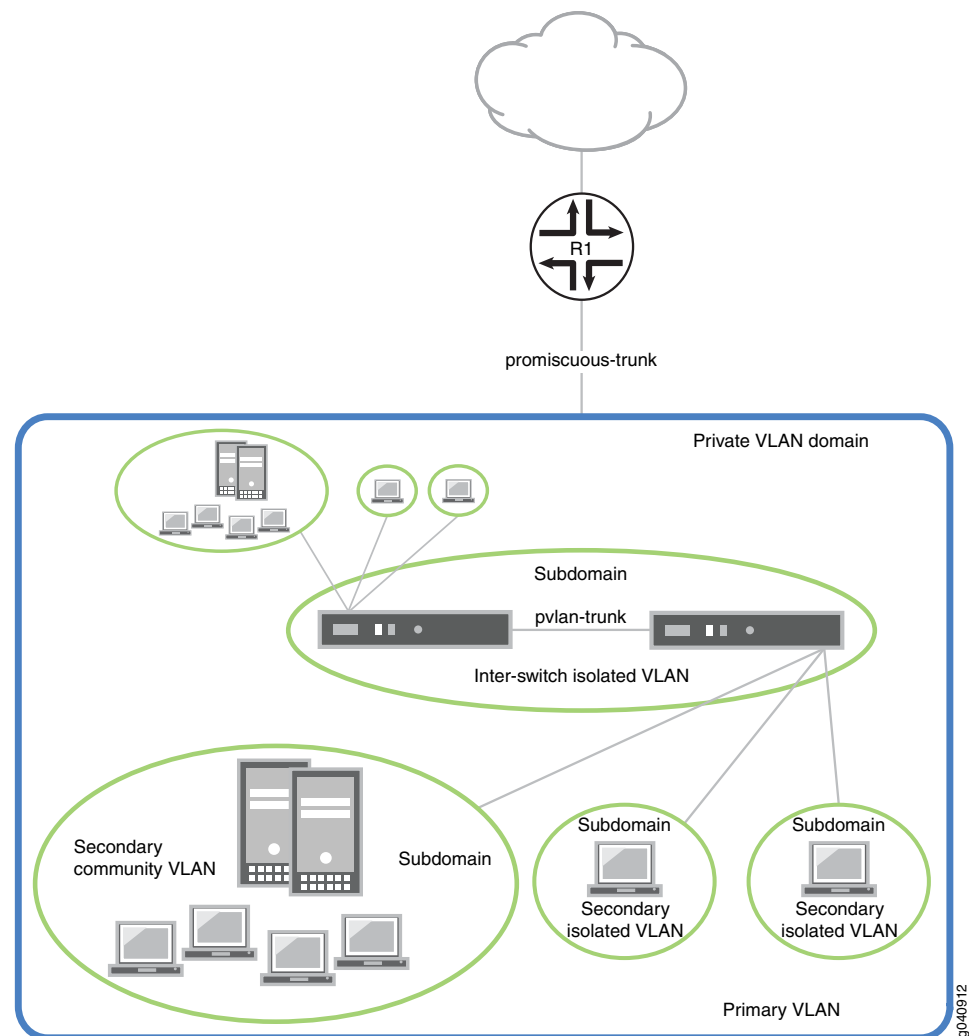
- [Typical Structure and Primary Application of PVLANS on page 190](#)
- [Using 802.1Q Tags to Identify Packets on page 191](#)
- [Efficient Use of IP Addresses on page 192](#)

- [PVLAN Port Types on page 192](#)
- [Limitations of Private VLANs on page 194](#)

Typical Structure and Primary Application of PVLANS

A PVLAN can be created on a single switch or can be configured to span multiple switches. The PVLAN shown in [Figure 10 on page 190](#) includes two switches, with a primary PVLAN domain and various subdomains.

Figure 10: Subdomains in a PVLAN



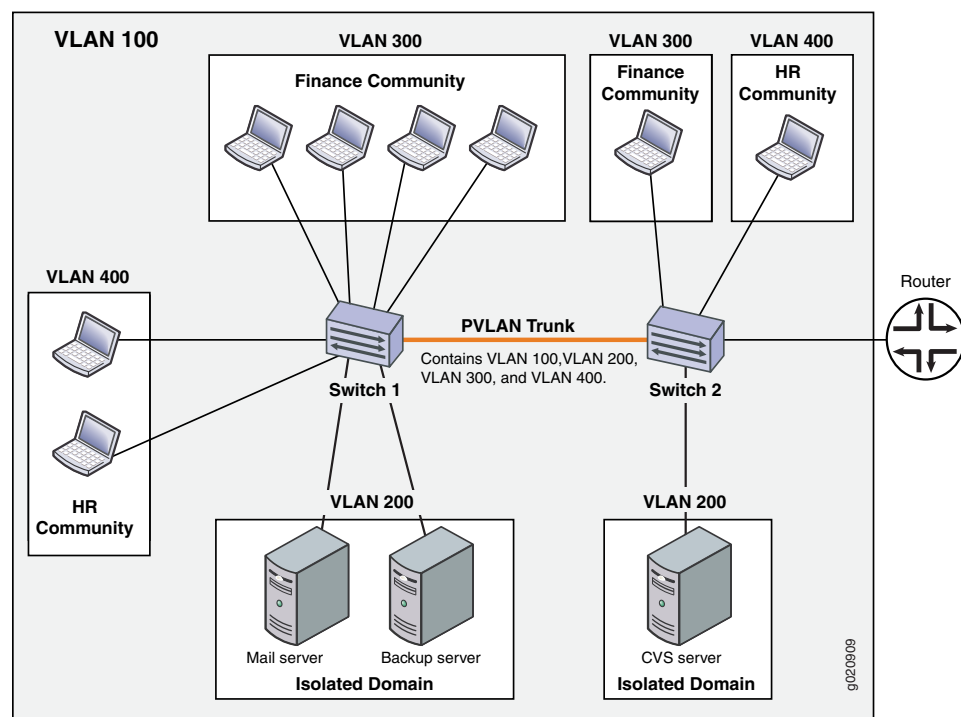
As shown in [Figure 10 on page 190](#), a PVLAN has only one primary domain and multiple secondary domains. The types of domains are:

- **Primary VLAN**—VLAN used to forward frames downstream to isolated and community VLANs.
- **Secondary isolated VLAN**—VLAN that receives packets only from the primary VLAN and forwards frames upstream to the primary VLAN.

- Secondary interswitch isolated VLAN—VLAN used to forward isolated VLAN traffic from one switch to another through PVLAN trunk ports. 802.1Q tags are required for interswitch isolated VLANs because IEEE 802.1Q uses an internal tagging mechanism by which a trunking device inserts a 4-byte VLAN frame identification tab into the packet header.
- Secondary community VLAN—VLAN used to transport frames among members of a community (a subset of users within the VLAN) and to forward frames upstream to the primary VLAN.

Figure 11 on page 191 shows a PVLAN spanning multiple switches, where the primary VLAN (100) contains two community domains (300 and 400) and one interswitch isolated domain.

Figure 11: PVLAN Spanning Multiple Switches



NOTE: Primary and secondary VLANs count against the limit of 4089 VLANs supported on the QFX Series. For example, each VLAN in Figure 11 on page 191 counts against this limit.

Using 802.1Q Tags to Identify Packets

When packets are marked with a customer-specific 802.1Q tag, that tag identifies ownership of the packets for any switch or router in the network. Sometimes, 802.1Q tags are needed within PVLANS to keep track of packets from different subdomains.

[Table 17 on page 192](#) indicates when a VLAN 802.1Q tag is needed on the primary VLAN or on secondary VLANs.

Table 17: PVLAN Requirements for 802.1Q Tags

	On a Single Switch	On Multiple Switches
Primary VLAN	Specify an 802.1Q tag by setting a VLAN ID.	Specify an 802.1Q tag by setting a VLAN ID.
Secondary VLAN	No tag needed on VLANs.	VLANs need 802.1Q tags: <ul style="list-style-type: none"> Specify an 802.1Q tag for each community VLAN by setting a VLAN ID. Specify the 802.1Q tag for an isolation VLAN ID by setting an isolation ID.

Efficient Use of IP Addresses

PVLANS provide IP address conservation and efficient allocation of IP addresses. In a typical network, VLANs usually correspond to a single IP subnet. In PVLANS, the hosts in all secondary VLANs belong to the same IP subnet because the subnet is allocated to the primary VLAN. Hosts within the secondary VLAN are assigned IP addresses based on IP subnets associated with the primary VLAN, and their IP subnet masking information reflects that of the primary VLAN subnet. However, each secondary VLAN is a separate broadcast domain.

PVLAN Port Types

PVLANS can use six different port types. The network depicted in [Figure 11 on page 191](#) uses a promiscuous port to transport information to the router, community ports to connect the finance and HR communities to their respective switches, isolated ports to connect the servers, and a PVLAN trunk port to connect the two switches. PVLAN ports have different restrictions:

- Promiscuous trunk port—A promiscuous port is an upstream trunk port connected to a router, firewall, server, or provider network. A promiscuous trunk port can communicate with all interfaces, including the isolated and community ports within a PVLAN.
- PVLAN trunk port—A PVLAN trunk port is required in multiswitch PVLAN configurations to span the switches. The PVLAN trunk port is a member of all VLANs within the PVLAN (that is, the primary VLAN, the community VLANs, and the interswitch isolated VLAN), and it carries traffic from the primary VLAN and all secondary VLANs. It can communicate with all ports.

Communication between a PVLAN trunk port and an isolated port is usually unidirectional. A PVLAN trunk port's membership in the interswitch isolated VLAN is egress-only, meaning that an isolated port can forward packets to a PVLAN trunk port, but a PVLAN trunk port does not forward packets to an isolated port (unless the packets ingress on a promiscuous access port and are therefore being forwarded to all the secondary VLANs in the same primary VLAN as the promiscuous port).

- **Secondary VLAN trunk port (not shown)**—Secondary trunk ports carry secondary VLAN traffic. For a given private VLAN, a secondary VLAN trunk port can carry traffic for only one secondary VLAN. However, a secondary VLAN trunk port can carry traffic for multiple secondary VLANs as long as each secondary VLAN is a member of a different primary VLAN. For example, a secondary VLAN trunk port can carry traffic for a community VLAN that is part of primary VLAN pvlan100 and also carry traffic for an isolated VLAN that is part of primary VLAN pvlan400.
- **Community port**—Community ports communicate among themselves and with their promiscuous ports. Community ports serve only a select group of users. These interfaces are separated at Layer 2 from all other interfaces in other communities or isolated ports within their PVLAN.
- **Isolated access port**—Isolated ports have Layer 2 connectivity only with promiscuous ports and PVLAN trunk ports—an isolated port cannot communicate with another isolated port even if these two ports are members of the same isolated VLAN (or interswitch isolated VLAN) domain. Typically, a server, such as a mail server or a backup server, is connected on an isolated port. In a hotel, each room would typically be connected on an isolated port, meaning that room-to-room communication is not possible, but each room can access the Internet on the promiscuous port.
- **Promiscuous access port (not shown)**—These ports carry untagged traffic. Traffic that ingresses on a promiscuous access port is forwarded to all secondary VLAN ports on the device. If traffic ingresses into the device on a VLAN-enabled port and egresses on a promiscuous access port, the traffic is untagged on egress. If tagged traffic ingresses on a promiscuous access port, the traffic is discarded.

Table 18 on page 193 summarizes whether Layer 2 connectivity exists between the different types of ports.

Table 18: PVLAN Ports and Layer 2 Connectivity

Port Type	Promiscuous Trunk	PVLAN Trunk	Secondary Trunk	Community	Isolated Access	Promiscuous access
Promiscuous trunk	Yes	Yes	Yes	Yes	Yes	Yes
PVLAN trunk	Yes	Yes	Yes	Yes—same community only	Yes	Yes
Secondary Trunk	Yes	Yes	No	Yes	No	Yes
Community	Yes	Yes	Yes	Yes—same community only	No	Yes
Isolated access	Yes	Yes—unidirectional only	No	No	No	Yes
Promiscuous access	Yes	Yes	Yes	Yes	Yes	No



NOTE: If you enable the `no-mac-learning` statement on a primary VLAN, all isolated VLANs in the PVLAN inherit that setting. However, if you want to disable MAC address learning on any community VLANs, you must configure the `no-mac-learning` statement on each of those VLANs.

Limitations of Private VLANs

The following constraints apply to private VLAN configurations:

- IGMP snooping is not supported with private VLANs.
- Routed VLAN interfaces are not supported on private VLANs
- Routing between secondary VLANs in the same primary VLAN is not supported.

Related Documentation

- [Using an IRB Interface in a Private VLAN on page 198](#)
- [Understanding Secondary VLAN Trunk Ports and Promiscuous Access Ports on PVLANS](#)
- [Creating a Private VLAN on a Single Switch](#)
- [Creating a Private VLAN Spanning Multiple Switches](#)

Understanding PVLAN Traffic Flows Across Multiple Switches

This topic illustrates and explains three different traffic flows on a sample multiswitch network configured with a private VLAN (PVLAN). PVLANS restrict traffic flows through their member switch ports (which are called “private ports”) so that they communicate only with a specific uplink trunk port or with specified ports within the same VLAN.

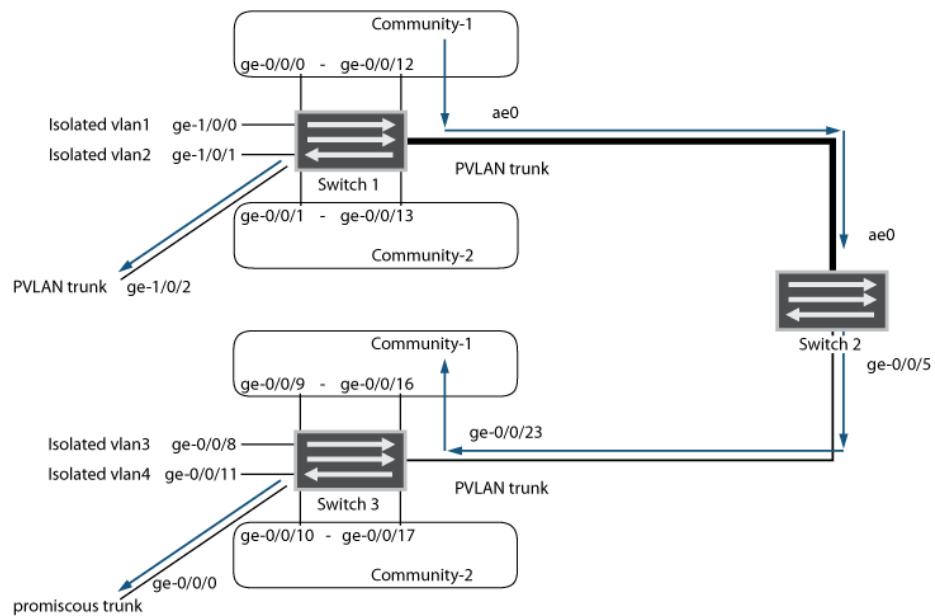
This topic describes:

- [Community VLAN Sending Untagged Traffic on page 194](#)
- [Isolated VLAN Sending Untagged Traffic on page 195](#)
- [PVLAN Tagged Traffic Sent on a Promiscuous Port on page 196](#)

Community VLAN Sending Untagged Traffic

In this scenario, a VLAN in Community-1 of Switch 1 at interface `ge-0/0/0` sends untagged traffic. The arrows in [Figure 12 on page 195](#) represent this traffic flow.

Figure 12: Community VLAN Sends Untagged Traffic



In this scenario, the following activity takes place on Switch 1:

- Community-1 VLAN on interface ge-0/0/0: Learning
- pvlan100 on interface ge-0/0/0: Replication
- Community-1 VLAN on interface ge-0/0/12: Receives traffic
- PVLAN trunk port: Traffic exits from ge-1/0/2 and from ae0 with tag 10
- Community-2: Interface receives no traffic
- Isolated VLANs: Interfaces receive no traffic

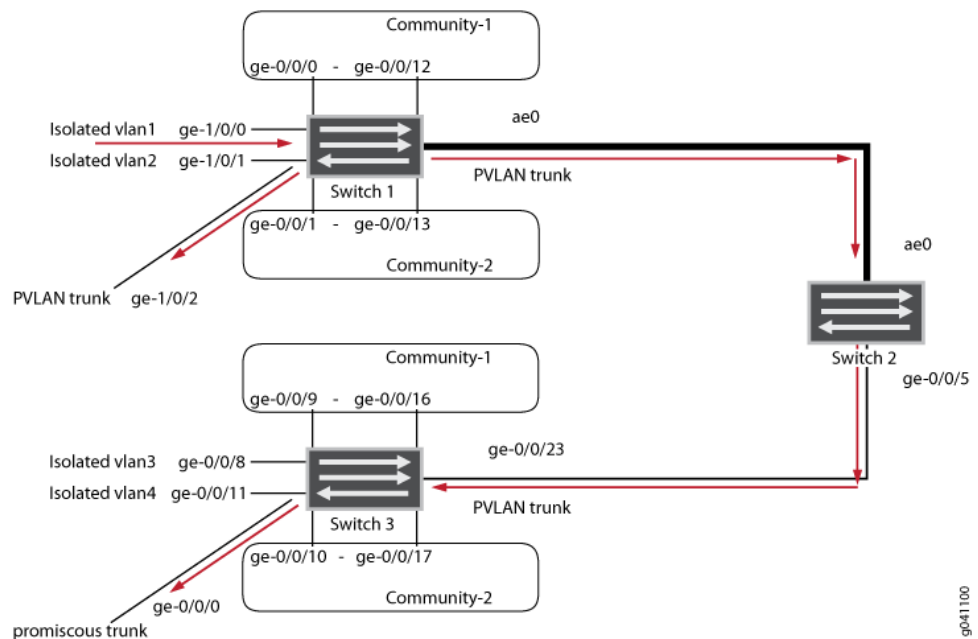
In this scenario, this activity takes place on Switch 3:

- Community-1 VLAN on interface ge-0/0/23 (PVLAN trunk): Learning
- pvlan100 on interface ge-0/0/23: Replication
- Community-1 VLAN on interface ge-0/0/9 and ge-0/0/16: Receives traffic
- Promiscuous trunk port: Traffic exits from ge-0/0/0 with tag 100
- Community-2: Interface receives no traffic
- Isolated VLANs: Interfaces receive no traffic

Isolated VLAN Sending Untagged Traffic

In this scenario, isolated VLAN1 on Switch 1 at interface ge-1/0/0 sends untagged traffic. The arrows in [Figure 13 on page 196](#) represent this traffic flow.

Figure 13: Isolated VLAN Sends Untagged Traffic



In this scenario, the following activity takes place on Switch 1:

- Isolated VLAN1 on interface ge-1/0/0: Learning
- pvlan100 on interface ge-1/0/0: Replication
- Traffic exits from pvlan-trunk ge-1/0/2 and ae0 with tag 50
- Community-1 and Community-2: Interfaces receive no traffic
- Isolated VLANs: Interfaces receive no traffic

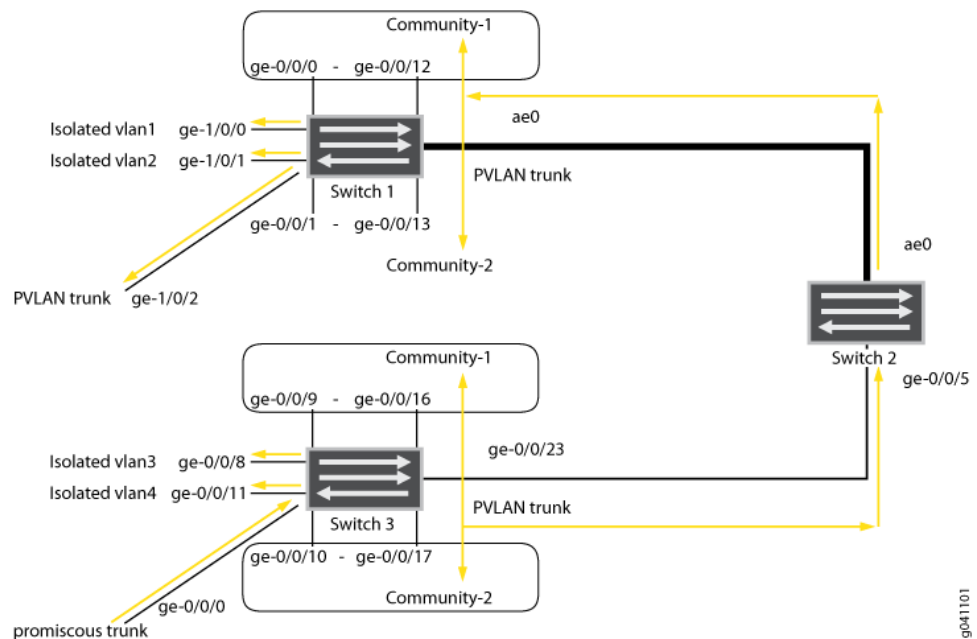
In this scenario, this activity takes place on Switch 3:

- VLAN on interface ge-0/0/23 (PVLAN trunk port): Learning
- pvlan100 on interface ge0/0/23: Replication
- Promiscuous trunk port: Traffic exits from ge-0/0/0 with tag 100
- Community-1 and Community-2: Interfaces receive no traffic
- Isolated VLANs: Receive no traffic

PVLAN Tagged Traffic Sent on a Promiscuous Port

In this scenario, PVLAN tagged traffic is sent on a promiscuous port. The arrows in [Figure 14 on page 197](#) represent this traffic flow.

Figure 14: PVLAN Tagged Traffic Sent on a Promiscuous Port



In this scenario, the following activity takes place on Switch 1:

- pvlan100 VLAN on interface ae0 (PVLAN trunk): Learning
- Community-1, Community-2, and all isolated VLANs on interface ae0: Replication
- VLAN on interface ae0: Replication
- Traffic exits from pvlan-trunk ge-1/0/2 with tag 100
- Community-1 and Community-2: Interfaces receive traffic
- Isolated VLANs: Receive traffic

In this scenario, this activity takes place on Switch 3:

- pvlan100 on interface ge-0/0/0: Learning
- Community-1, Community-2 and all isolated VLANs on interface ge-0/0/0: Replication
- VLAN on interface ge-0/0/0: Replication
- Community-1 and Community-2: Interfaces receive traffic
- Isolated VLANs: Receive traffic

Related Documentation

- [Understanding Private VLANs on EX Series Switches](#)
- [Example: Configuring a Private VLAN on a Single EX Series Switch](#)
- [Example: Configuring a Private VLAN Spanning Multiple EX Series Switches](#)
- [Creating a Private VLAN on a Single EX Series Switch \(CLI Procedure\)](#)
- [Creating a Private VLAN Spanning Multiple EX Series Switches \(CLI Procedure\)](#)

- [Understanding Private VLANs on page 189](#)
- [Creating a Private VLAN on a Single Switch](#)
- [Creating a Private VLAN Spanning Multiple Switches](#)
- [Example: Configuring a Private VLAN on a Single Switch](#)
- [Example: Configuring a Private VLAN Spanning Multiple Switches](#)

Using an IRB Interface in a Private VLAN

VLANs limit broadcasts to specified users. Private VLANs (PVLANS) take this concept a step further by splitting the broadcast domain into multiple isolated broadcast subdomains and essentially putting secondary VLANs inside a primary VLAN. PVLANS restrict traffic flows through their member switch ports (called “private ports”) so that these ports communicate only with a specified uplink trunk port or with specified ports within the same VLAN. PVLANS are useful for restricting the flow of broadcast and unknown unicast traffic and for limiting the communication between known hosts. Service providers use PVLANS to keep their customers isolated from one another.

Just like regular VLANs, PVLANS are isolated at Layer 2 and normally require that a Layer 3 device be used if you want to route traffic. Starting with Junos OS 14.1X53-D30, you can use an integrated routing and bridging (IRB) interface to route Layer 3 traffic between devices connected to a PVLAN. Using an IRB interface in this way can also allow the devices in the PVLAN to communicate at Layer 3 with devices outside the PVLAN.

- [Configuring an IRB Interface in a Private VLAN on page 198](#)
- [IRB Interface Limitation in a PVLAN on page 198](#)

Configuring an IRB Interface in a Private VLAN

Use the following guidelines when configuring an IRB interface in a PVLAN:

- You can create only one IRB interface in a PVLAN, regardless of how many switches participate in the PVLAN.
- The IRB interface must be a member of the primary VLAN in the PVLAN.
- Each host device that you want to connect at Layer 3 must use the IP address of the IRB as its default gateway address.
- Because the devices in the community and isolated VLANs are isolated at Layer 2, you must configure the following statement for the IRB interface to allow ARP resolution to occur between the VLANs so that devices can communicate at Layer 3:

set interfaces irb unit *unit-number* proxy-arp unrestricted

IRB Interface Limitation in a PVLAN

If your PVLAN includes multiple switches, an issue can occur if the Ethernet switching table is cleared on a switch that does not have an IRB interface. If a Layer 3 packet transits

the switch before its destination MAC address is learned again, it is broadcast to all the Layer 3 hosts connected to the PVLAN.

Related Documentation

- [Understanding Private VLANs on page 189](#)
- [Configuring IRB Interfaces on page 72](#)
- [Creating a Private VLAN on a Single Switch](#)
- [Example: Configuring a Private VLAN Spanning Multiple Switches With an IRB Interface on page 199](#)

Example: Configuring a Private VLAN Spanning Multiple Switches With an IRB Interface

For security reasons, it is often useful to restrict the flow of broadcast and unknown unicast traffic and even to limit the communication between known hosts. The private VLAN (PVLAN) feature allows an administrator to split a broadcast domain into multiple isolated broadcast subdomains, essentially putting a VLAN inside a VLAN. A PVLAN can span multiple switches. This example describes how to create a PVLAN spanning multiple switches. The example creates one primary PVLAN, containing multiple secondary VLANs.

Just like regular VLANs, PVLANS are isolated at Layer 2 and normally require that a Layer 3 device be used if you want to route traffic. Starting with Junos OS 14.1X53-D30, you can use an integrated routing and bridging (IRB) interface to route Layer 3 traffic between devices connected to a PVLAN. Using an IRB interface in this way can also allow the devices in the PVLAN to communicate at Layer 3 with devices in other community or isolated VLANs or with devices outside the PVLAN. This example also demonstrates how to include an IRB interface in a PVLAN configuration.

- [Requirements on page 199](#)
- [Overview and Topology on page 199](#)
- [Configuration Overview on page 202](#)
- [Configuring a PVLAN on Switch 1 on page 202](#)
- [Configuring a PVLAN on Switch 2 on page 204](#)
- [Configuring a PVLAN on Switch 3 on page 206](#)
- [Verification on page 208](#)

Requirements

This example uses the following hardware and software components:

- Three QFX switches
- Junos OS Release 14.1X53-D30 for the QFX Series

Overview and Topology

In a large office with multiple buildings and VLANs, you might need to isolate some workgroups or other endpoints for security reasons or to partition the broadcast domain. This configuration example shows how to create a PVLAN spanning multiple QFX

switches, with one primary VLAN containing two community VLANs (one for HR and one for Finance), and an interswitch isolated VLAN (for the mail server, the backup server, and the CVS server). The PVLAN comprises three switches—two access switches and one distribution switch. The devices in the PVLAN are connected at Layer 3 to each other and to devices outside the PVLAN through an IRB interface configured on the distribution switch.



NOTE: The isolated ports on Switch 1 and on Switch 2 do not have Layer 2 connectivity with one another even though they are included within the same domain. See “Understanding Private VLANs” on page 189.

Figure 15 on page 200 shows the topology for this example.

Figure 15: PVLAN Topology Spanning Multiple Switches with an IRB Interface

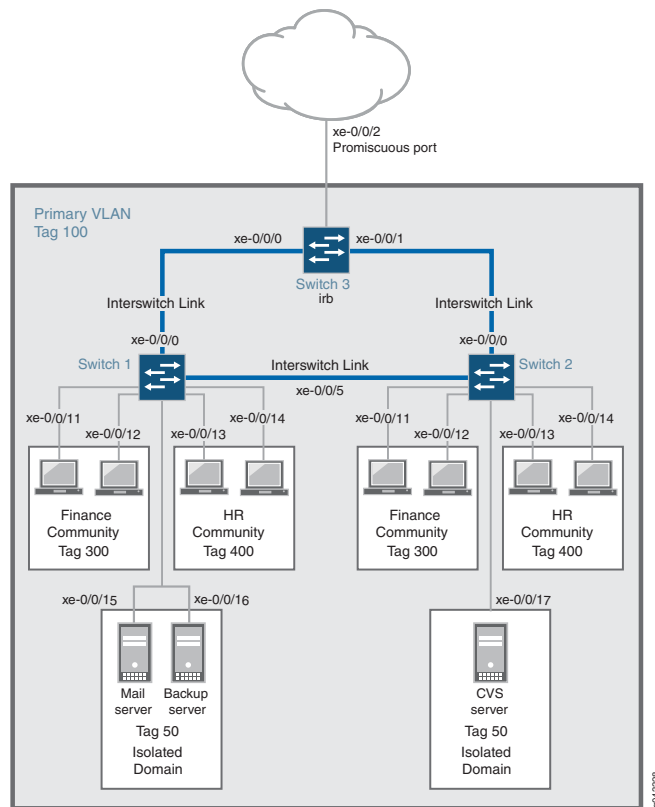


Table 19 on page 201, Table 20 on page 201, and Table 21 on page 202 list the settings for the example topology.

Table 19: Components of Switch 1 in the Topology for Configuring a PVLAN Spanning Multiple Devices

Property	Settings
VLAN names and tag IDs	primary-vlan , tag 100 isolated-vlan-id , tag 50 finance-comm , tag 300 hr-comm , tag 400
Interswitch link interfaces	xe-0/0/0.0 , connects Switch 1 to Switch 3 xe-0/0/5.0 , connects Switch 1 to Switch 2
Isolated Interfaces in primary VLAN	xe-0/0/15.0 , mail server xe-0/0/16.0 , backup server
Interfaces in VLAN finance-com	xe-0/0/11.0 xe-0/0/12.0
Interfaces in VLAN hr-comm	xe-0/0/13.0 xe-0/0/14.0

Table 20: Components of Switch 2 in the Topology for Configuring a PVLAN Spanning Multiple Devices

Property	Settings
VLAN names and tag IDs	primary-vlan , tag 100 isolated-vlan-id , tag 50 finance-comm , tag 300 hr-comm , tag 400
Interswitch link interfaces	xe-0/0/0.0 , connects Switch 2 to Switch 3 xe-0/0/5.0 , connects Switch 2 to Switch 1
Isolated Interface in primary VLAN	xe-0/0/17.0 , CVS server
Interfaces in VLAN finance-com	xe-0/0/11.0 xe-0/0/12.0
Interfaces in VLAN hr-comm	xe-0/0/13.0 xe-0/0/14.0

Table 21: Components of Switch 3 in the Topology for Configuring a PVLAN Spanning Multiple Devices

Property	Settings
VLAN names and tag IDs	primary-vlan , tag 100 isolated-vlan-id , tag 50 finance-comm , tag 300 hr-comm , tag 400
Interswitch link interfaces	xe-0/0/0.0 , connects Switch 3 to Switch 1. xe-0/0/1.0 , connects Switch 3 to Switch 2.
Promiscuous port	xe-0/0/2 , connects the PVLAN to another network. NOTE: You must configure the trunk port that connects the PVLAN to another switch or router outside the PVLAN as a member of the PVLAN, which implicitly configures it as a promiscuous port.
IRB interface	xe-0/0/0 xe-0/0/1 Configure unrestricted proxy ARP on the IRB interface to allow ARP resolution to occur so that devices that use IPv4 can communicate at Layer 3. For IPv6 traffic, you must explicitly map an IRB address to the destination address to allow ARP resolution.

Configuration Overview

When configuring a PVLAN on multiple switches, the following rules apply:

- The primary VLAN must be a tagged VLAN.
- The primary VLAN is the only VLAN that can be a member of an interswitch link interface.

When configuring an IRB interface in a PVLAN, these rules apply:

- You can create only one IRB interface in a PVLAN, regardless of how many switches participate in the PVLAN.
- The IRB interface must be a member of the primary VLAN in the PVLAN.
- Each host device that you want to connect at Layer 3 must use an IP address of the IRB as its default gateway address.

Configuring a PVLAN on Switch 1

CLI Quick Configuration

To quickly create and configure a PVLAN spanning multiple switches, copy the following commands and paste them into the terminal window of Switch 1:

```
[edit]
set interfaces xe-0/0/0 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/0 unit 0 family ethernet-switching inter-switch-link
set interfaces xe-0/0/0 unit 0 family ethernet-switching vlan members 100
set interfaces xe-0/0/5 unit 0 family ethernet-switching interface-mode trunk
```



```

set interfaces xe-0/0/5 unit 0 family ethernet-switching inter-switch-link
set interfaces xe-0/0/5 unit 0 family ethernet-switching vlan members 100
set vlans finance-comm vlan-id 300 private-vlan community
set vlans hr-comm vlan-id 400 private-vlan community
set vlans isolated-vlan vlan-id 50 private-vlan isolated
set vlans pvlan100 vlan-id 100 community-vlans [300 400] isolated-vlan 50
set interfaces xe-0/0/11 unit 0 family ethernet-switching vlan members 300
set interfaces xe-0/0/12 unit 0 family ethernet-switching vlan members 300
set interfaces xe-0/0/13 unit 0 family ethernet-switching vlan members 400
set interfaces xe-0/0/14 unit 0 family ethernet-switching vlan members 400
set interfaces xe-0/0/15 unit 0 family ethernet-switching vlan members 50
set interfaces xe-0/0/16 unit 0 family ethernet-switching vlan members 50

```

Step-by-Step Procedure

1. Configure interface xe-0/0/0 to be a trunk:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching interface-mode trunk
```
2. Configure interface xe-0/0/0 to be an interswitch link that carries all the VLANs:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching inter-switch-link
```
3. Configure pvlan100 (the primary VLAN) to be a member of interface xe-0/0/0:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching vlan members 100
```
4. Configure interface xe-0/0/5 to be a trunk:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching interface-mode trunk
```
5. Configure interface xe-0/0/5 to be an interswitch link that carries all the VLANs:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching inter-switch-link
```
6. Configure pvlan100 to be a member of interface xe-0/0/5:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching vlan members 100
```
7. Create the community VLAN for the finance organization:

```
[edit vlans]
set finance-comm vlan-id 300 private-vlan community
```
8. Create the community VLAN for the HR organization:

```
[edit vlans]
set hr-comm vlan-id 400 private-vlan community
```
9. Create the isolated VLAN for the mail and backup servers:

```
[edit vlans]
set isolated-vlan vlan-id 50 private-vlan isolated
```
10. Create the primary VLAN and make the community and isolated VLANs members of it:

```
[edit vlans]
set pvlan100 vlan-id 100 community-vlans [300 400] isolated-vlan 50
```
11. Configure VLAN 300 (the a community VLAN) to be a member of interface xe-0/0/11:

```
[edit interfaces]
user@switch# set xe-0/0/11 unit 0 family ethernet-switching vlan members 300
```
12. Configure VLAN 300 (a community VLAN) to be a member of interface xe-0/0/12:

```
[edit interfaces]
user@switch# set xe-0/0/12 unit 0 family ethernet-switching vlan members 300
```

13. Configure VLAN 400 (a community VLAN) to be a member of interface xe-0/0/13:

```
[edit interfaces]
user@switch# set xe-0/0/13 unit 0 family ethernet-switching vlan members 400
```
14. Configure VLAN 400 (a community VLAN) to be a member of interface xe-0/0/14:

```
[edit interfaces]
user@switch# set xe-0/0/14 unit 0 family ethernet-switching vlan members 400
```
15. Configure VLAN 50 (the isolated VLAN) to be a member of interface xe-0/0/15:

```
[edit interfaces]
user@switch# set xe-0/0/15 unit 0 family ethernet-switching vlan members 50
```
16. Configure VLAN 50 (the isolated VLAN) to be a member of interface xe-0/0/16:

```
[edit interfaces]
user@switch# set xe-0/0/16 unit 0 family ethernet-switching vlan members 50
```

Results

Check the results of the configuration:

```
[edit]
user@switch# show
vllans {
  finance-comm {
    vlan-id 300;
    private-vlan community;
  }
  hr-comm {
    vlan-id 400;
    private-vlan community;
  }
  isolated-vlan {
    vlan-id 50;
    private-vlan isolated;
  }
  pvlan100 {
    vlan-id 100;
    isolated-vlan 50;
    community-vllans [300 400]
  }
}
```

Configuring a PVLAN on Switch 2

CLI Quick Configuration To quickly create and configure a private VLAN spanning multiple switches, copy the following commands and paste them into the terminal window of Switch 2:



NOTE: The configuration of Switch 2 is the same as the configuration of Switch 1 except for the isolated VLAN. For Switch 2, the isolated VLAN interface is xe-0/0/17.0 .

```
[edit]
set interfaces xe-0/0/0 unit 0 family ethernet-switching interface-mode trunk
```

```

set interfaces xe-0/0/0 unit 0 family ethernet-switching inter-switch-link
set interfaces xe-0/0/0 unit 0 family ethernet-switching vlan members 100
set interfaces xe-0/0/5 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/5 unit 0 family ethernet-switching inter-switch-link
set interfaces xe-0/0/5 unit 0 family ethernet-switching vlan members 100
set vlans finance-comm vlan-id 300 private-vlan community
set vlans hr-comm vlan-id 400 private-vlan community
set vlans isolated-vlan vlan-id 50 private-vlan isolated
set vlans pvlan100 vlan-id 100 community-vlans [300 400] isolated-vlan 50
set interfaces xe-0/0/11 unit 0 family ethernet-switching vlan members 300
set interfaces xe-0/0/12 unit 0 family ethernet-switching vlan members 300
set interfaces xe-0/0/13 unit 0 family ethernet-switching vlan members 400
set interfaces xe-0/0/14 unit 0 family ethernet-switching vlan members 400
set interfaces xe-0/0/17 unit 0 family ethernet-switching vlan members 50

```

Step-by-Step Procedure

1. Configure interface xe-0/0/0 to be a trunk:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching interface-mode trunk
```
2. Configure interface xe-0/0/0 to be an interswitch link that carries all the VLANs:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching inter-switch-link
```
3. Configure pvlan100 (the primary VLAN) to be a member of interface xe-0/0/0:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching vlan members 100
```
4. Configure interface xe-0/0/5 to be a trunk:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching interface-mode trunk
```
5. Configure interface xe-0/0/5 to be an interswitch link that carries all the VLANs:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching inter-switch-link
```
6. Configure pvlan100 to be a member of interface xe-0/0/5:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching vlan members 100
```
7. Create the community VLAN for the finance organization:

```
[edit vlans]
set finance-comm vlan-id 300 private-vlan community
```
8. Create the community VLAN for the HR organization:

```
[edit vlans]
set hr-comm vlan-id 400 private-vlan community
```
9. Create the isolated VLAN for the mail and backup servers:

```
[edit vlans]
set isolated-vlan vlan-id 50 private-vlan isolated
```
10. Create the primary VLAN and make the community and isolated VLANs members of it:

```
[edit vlans]
set pvlan100 vlan-id 100 community-vlans [300 400] isolated-vlan 50
```
11. Configure VLAN 300 (the a community VLAN) to be a member of interface xe-0/0/11:

```
[edit interfaces]
user@switch# set xe-0/0/11 unit 0 family ethernet-switching vlan members 300
```
12. Configure VLAN 300 (a community VLAN) to be a member of interface xe-0/0/12:

- ```
[edit interfaces]
user@switch# set xe-0/0/12 unit 0 family ethernet-switching vlan members 300
```
13. Configure VLAN 400 (a community VLAN) to be a member of interface xe-0/0/13:
- ```
[edit interfaces]
user@switch# set xe-0/0/13 unit 0 family ethernet-switching vlan members 400
```
14. Configure VLAN 400 (a community VLAN) to be a member of interface xe-0/0/14:
- ```
[edit interfaces]
user@switch# set xe-0/0/14 unit 0 family ethernet-switching vlan members 400
```
15. Configure VLAN 50 (the isolated VLAN) to be a member of interface xe-0/0/17:
- ```
[edit interfaces]
user@switch# set xe-0/0/17 unit 0 family ethernet-switching vlan members 50
```

Results

Check the results of the configuration:

```
[edit]
user@switch# show
vllans {
  finance-comm {
    vlan-id 300;
    private-vlan community;
  }
  hr-comm {
    vlan-id 400;
    private-vlan community;
  }
  isolated-vlan {
    vlan-id 50;
    private-vlan isolated;
  }
  pvlan100 {
    vlan-id 100;
    isolated-vlan 50;
    community-vllans [300 400]
  }
}
```

Configuring a PVLAN on Switch 3

CLI Quick Configuration To quickly configure Switch 3 to function as the distribution switch of this PVLAN, copy the following commands and paste them into the terminal window of Switch 3:



NOTE: Interface xe-0/0/2.0 is a trunk port connecting the PVLAN to another network.

```
[edit]
[edit]
set interfaces xe-0/0/0 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/0 unit 0 family ethernet-switching inter-switch-link
set interfaces xe-0/0/0 unit 0 family ethernet-switching vlan members 100
```

```

set interfaces xe-0/0/1 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/1 unit 0 family ethernet-switching inter-switch-link
set interfaces xe-0/0/1 unit 0 family ethernet-switching vlan members 100
set interfaces xe-0/0/2 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/2 unit 0 family ethernet-switching vlan members 100
set vlans pvlan100 vlan-id 100
set interfaces irb unit 100 family inet address 192.168.1.1/24
set vlans pvlan100 l3-interface irb.100
set interfaces irb unit 100 proxy-arp unrestricted

```

Step-by-Step Procedure To configure Switch 3 to function as the distribution switch for this PVLAN, use the following procedure:

1. Configure interface xe-0/0/0 to be a trunk:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching interface-mode trunk
```
2. Configure interface xe-0/0/0 to be an interswitch link that carries all the VLANs:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching inter-switch-link
```
3. Configure pvlan100 (the primary VLAN) to be a member of interface xe-0/0/0:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching vlan members 100
```
4. Configure interface xe-0/0/5 to be a trunk:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching interface-mode trunk
```
5. Configure interface xe-0/0/5 to be an interswitch link that carries all the VLANs:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching inter-switch-link
```
6. Configure pvlan100 to be a member of interface xe-0/0/5:

```
[edit interfaces]
user@switch# set xe-0/0/5 unit 0 family ethernet-switching vlan members 100
```
7. Configure interface xe-0/0/2 (the promiscuous interface) to be a trunk:

```
[edit interfaces]
user@switch# set xe-0/0/2 unit 0 family ethernet-switching interface-mode trunk
```
8. Configure pvlan100 to be a member of interface xe-0/0/2:

```
[edit interfaces]
user@switch# set xe-0/0/2 unit 0 family ethernet-switching vlan members 100
```
9. Create the primary VLAN:

```
[edit vlans]
set vlans pvlan100 vlan-id 100
```
10. Create the IRB interface **irb** and assign it an address in the subnet used by the devices attached to Switches 1 and 2:

```
[edit interfaces]
set irb unit 100 family inet address 192.168.1.1/24
```



NOTE: Each host device that you want to connect at Layer 3 must be in the same subnet as the IRB interface and use the IP address of the IRB interface as its default gateway address.

11. Complete the IRB interface configuration by binding the interface to the primary VLAN **pvlan100**:

[edit vlans]
set pvlan100 l3-interface irb.100
12. Configure unrestricted proxy ARP for each unit of the IRB interface so that ARP resolution works for IPv4 traffic:

[edit interfaces]
set irb unit 100 proxy-arp unrestricted



NOTE: Because the devices in the community and isolated VLANs are isolated at Layer 2, this step is required to allow ARP resolution to occur between the VLANs so that devices using IPv4 can communicate at Layer 3. (For IPv6 traffic, you must explicitly map an IRB address to the destination address to allow ARP resolution.)

Results

Check the results of the configuration:

```
[edit]
user@switch# show
vlans {
  pvlan100{
    vlan-id 100;
  }
}
```

Verification

To confirm that the configuration is working properly, perform these tasks:

- [Verifying That the Primary VLAN and Secondary VLANs Were Created on Switch 1 on page 208](#)
- [Verifying That the Primary VLAN and Secondary VLANs Were Created on Switch 2 on page 210](#)
- [Verifying That the Primary VLAN and Secondary VLANs Were Created on Switch 3 on page 211](#)

Verifying That the Primary VLAN and Secondary VLANs Were Created on Switch 1

Purpose Verify that the PVLAN configuration spanning multiple switches is working properly on Switch 1:

Action Use the **show vlans extensive** command:

```
user@switch> show vlans extensive
VLAN: __pvlan_pvlan100_xe-0/0/15.0__, Created at: Wed Sep 16 23:15:27 2015
Internal index: 5, Admin State: Enabled, Origin: Static
```

```

Private VLAN Mode: Isolated, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 1 (Active = 1)
    xe-0/0/0.0*, tagged, trunk    xe-0/0/5.0*, tagged, trunk    xe-0/0/15.0*,
    untagged, access

VLAN: __pvlan_pvlan100_xe-0/0/16.0__, Created at: Wed Sep 16 23:15:27 2015
Internal index: 6, Admin State: Enabled, Origin: Static
Private VLAN Mode: Isolated, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 1 (Active = 1)
    xe-0/0/0.0*, tagged, trunk    xe-0/0/5.0*, tagged, trunk    xe-0/0/16.0*,
    untagged, access

VLAN: __pvlan_pvlan100_isiv__, Created at: Wed Sep 16 23:15:27 2015
802.1Q Tag: 50, Internal index: 7, Admin State: Enabled, Origin: Static
Private VLAN Mode: Inter-switch-isolated, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 0 (Active = 0)
    xe-0/0/0.0*, tagged, trunk    xe-0/0/5.0*, tagged, trunk
VLAN: default, Created at: Wed Sep 16 03:03:18 2015
Internal index: 2, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 0 (Active = 0), Untagged 0 (Active = 0)

VLAN: finance-comm, Created at: Wed Sep 16 23:15:27 2015
802.1Q Tag: 300, Internal index: 8, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 2 (Active = 2)
    xe-0/0/0.0*, tagged, trunk    xe-0/0/5.0*, tagged, trunk    xe-0/0/11.0*,
    untagged, access
    xe-0/0/12.0*, untagged, access

VLAN: hr-comm, Created at: Wed Sep 16 23:15:27 2015
802.1Q Tag: 400, Internal index: 9, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 2 (Active = 2)
    xe-0/0/0.0*, tagged, trunk    xe-0/0/5.0*, tagged, trunk    xe-0/0/13.0*,
    untagged, access
    xe-0/0/14.0*, untagged, access

VLAN: pvlan100, Created at: Wed Sep 16 23:15:27 2015
802.1Q Tag: 100, Internal index: 4, Admin State: Enabled, Origin: Static
Private VLAN Mode: Primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 6 (Active = 6)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/5.0*, tagged, trunk    xe-0/0/11.0*, untagged, access
    xe-0/0/12.0*, untagged, access
    xe-0/0/13.0*, untagged, access
    xe-0/0/14.0*, untagged, access
    xe-0/0/15.0*, untagged, access
    xe-0/0/16.0*, untagged, access
Secondary VLANs: Isolated 2, Community 2, Inter-switch-isolated 1
Isolated VLANs :
    __pvlan_pvlan100_xe-0/0/15.0__
    __pvlan_pvlan100_xe-0/0/16.0__
Community VLANs :
    finance-comm

```

```

hr-comm
Inter-switch-isolated VLAN :
__pvlan_pvlan100_isiv__

```

Meaning The output shows that a PVLAN was created on Switch 1 and shows that it includes two isolated VLANs, two community VLANs, and an interswitch isolated VLAN. The presence of the trunk and Inter-switch-isolated fields indicates that this PVLAN is spanning more than one switch.

Verifying That the Primary VLAN and Secondary VLANs Were Created on Switch 2

Purpose Verify that the PVLAN configuration spanning multiple switches is working properly on Switch 2:

Action Use the `show vlans extensive` command:

```

user@switch> show vlans extensive
VLAN: __pvlan_pvlan100_xe-0/0/17.0__, Created at: Wed Sep 16 23:19:22 2015
Internal index: 5, Admin State: Enabled, Origin: Static
Private VLAN Mode: Isolated, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 1 (Active = 1)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/5.0*, tagged, trunk
    xe-0/0/17.0*, untagged, access

VLAN: __pvlan_pvlan100_isiv__, Created at: Wed Sep 16 23:19:22 2015
802.1Q Tag: 50, Internal index: 6, Admin State: Enabled, Origin: Static
Private VLAN Mode: Inter-switch-isolated, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 0 (Active = 0)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/5.0*, tagged, trunk

VLAN: default, Created at: Wed Sep 16 03:03:18 2015
Internal index: 2, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 0 (Active = 0), Untagged 0 (Active = 0)

VLAN: finance-comm, Created at: Wed Sep 16 23:19:22 2015
802.1Q Tag: 300, Internal index: 7, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 2 (Active = 2)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/5.0*, tagged, trunk
    xe-0/0/11.0*, untagged, access
    xe-0/0/12.0*, untagged, access

VLAN: hr-comm, Created at: Wed Sep 16 23:19:22 2015
802.1Q Tag: 400, Internal index: 8, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 2 (Active = 2)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/5.0*, tagged, trunk
    xe-0/0/13.0*, untagged, access

```



```
xe-0/0/14.0*, untagged, access
```

```
VLAN: pvlan100, Created at: Wed Sep 16 23:19:22 2015
802.1Q Tag: 100, Internal index: 4, Admin State: Enabled, Origin: Static
Private VLAN Mode: Primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 5 (Active = 5)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/5.0*, tagged, trunk
    xe-0/0/11.0*, untagged, access
    xe-0/0/12.0*, untagged, access
    xe-0/0/13.0*, untagged, access
    xe-0/0/14.0*, untagged, access
    xe-0/0/17.0*, untagged, access
Secondary VLANs: Isolated 1, Community 2, Inter-switch-isolated 1
Isolated VLANs :
    __pvlan_pvlan100_xe-0/0/17.0__
Community VLANs :
    finance-comm
    hr-comm
Inter-switch-isolated VLAN :
    __pvlan_pvlan100_isiv__
```

Meaning The output shows that a PVLAN was created on Switch 2 and shows that it includes one isolated VLAN, two community VLANs, and an interswitch isolated VLAN. The presence of the trunk and Inter-switch-isolated fields indicates that this PVLAN is spanning more than one switch. When you compare this output to the output of Switch 1, you can see that both switches belong to the same PVLAN (**pvlan100**).

Verifying That the Primary VLAN and Secondary VLANs Were Created on Switch 3

Purpose Verify that the PVLAN configuration spanning multiple switches is working properly on Switch 3:

Action Use the **show vlans extensive** command:

```
user@switch> show vlans extensive
VLAN: __pvlan_pvlan100_isiv__, Created at: Wed Sep 16 23:22:40 2015
802.1Q Tag: 50, Internal index: 5, Admin State: Enabled, Origin: Static
Private VLAN Mode: Inter-switch-isolated, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 0 (Active = 0)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/1.0*, tagged, trunk

VLAN: default, Created at: Wed Sep 16 03:03:18 2015
Internal index: 2, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 0 (Active = 0), Untagged 0 (Active = 0)

VLAN: finance-comm, Created at: Wed Sep 16 23:22:40 2015
802.1Q Tag: 300, Internal index: 6, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 0 (Active = 0)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/1.0*, tagged, trunk
```

```

VLAN: hr-comm, Created at: Wed Sep 16 23:22:40 2015
802.1Q Tag: 400, Internal index: 7, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: pvlan100
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 0 (Active = 0)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/1.0*, tagged, trunk

VLAN: pvlan100, Created at: Wed Sep 16 23:22:40 2015
802.1Q Tag: 100, Internal index: 4, Admin State: Enabled, Origin: Static
Private VLAN Mode: Primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged 0 (Active = 0)
    xe-0/0/0.0*, tagged, trunk
    xe-0/0/1.0*, tagged, trunk
Secondary VLANs: Isolated 0, Community 2, Inter-switch-isolated 1
Community VLANs :
    finance-comm
    hr-comm
Inter-switch-isolated VLAN :
    __pvlan_pvlan100_isiv__

```

Meaning The output shows that the PVLAN (**pvlan100**) is configured on Switch 3 and that it includes no isolated VLANs, two community VLANs, and an interswitch isolated VLAN. But Switch 3 is functioning as a distribution switch, so the output does not include access interfaces within the PVLAN. It shows only the trunk interfaces that connect **pvlan100** from Switch 3 to the other switches (Switch 1 and Switch 2) in the same PVLAN.

- Related Documentation**
- [Understanding Private VLANs on page 189](#)
 - [Using an IRB Interface in a Private VLAN on page 198](#)
 - [Understanding PVLAN Traffic Flows Across Multiple Switches on page 194](#)
 - [Example: Configuring a Private VLAN on a Single Switch](#)

Understanding Egress Firewall Filters with PVLANS

If you apply firewall filters to private VLANs in the output direction, the behavior of the filters might be unexpected. This topic explains how egress filters behave when applied to private VLANs.

If you apply a firewall filter in the output direction to a primary VLAN, the filter also applies to the secondary VLANs that are members of the primary VLAN when the traffic egresses with the primary VLAN tag or isolated VLAN tag, as listed below:

- Traffic forwarded from a secondary VLAN trunk port to a promiscuous port (trunk or access)
- Traffic forwarded from a secondary VLAN trunk port to a PVLAN trunk port.
- Traffic forwarded from a promiscuous port (trunk or access) to a secondary VLAN trunk port

- Traffic forwarded from a PVLAN trunk port. to a secondary VLAN trunk port
- Traffic forwarded from a community port to a promiscuous port (trunk or access)

If you apply a firewall filter in the output direction to a primary VLAN, the filter does *not* apply to traffic that egresses with a community VLAN tag, as listed below:

- Traffic forwarded from a community trunk port to a PVLAN trunk port
- Traffic forwarded from a promiscuous port (trunk or access) to a community trunk port
- Traffic forwarded from a PVLAN trunk port. to a community trunk port

If you apply a firewall filter in the output direction to a community VLAN, the following behaviors apply:

- The filter is applied to traffic forwarded from a promiscuous port (trunk or access) to a community trunk port (because the traffic egresses with the community VLAN tag).
- The filter is applied to traffic forwarded from a community port to a PVLAN trunk port (because the traffic egresses with the community VLAN tag).
- The filter is *not* applied to traffic forwarded from a community port to a promiscuous port (because the traffic egresses with the primary VLAN tag or untagged).

**Related
Documentation**

- [Understanding Private VLANs on page 189](#)
- *Example: Configuring PVLANs with Secondary VLAN Trunk Ports and Promiscuous Access Ports*
- *Creating a Private VLAN on a Single Switch*
- *Creating a Private VLAN Spanning Multiple Switches*
- *Troubleshooting Private VLANs*

Creating a Private VLAN on a Single Switch (CLI Procedure)



NOTE: This task uses Junos OS for switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your EX Series switch runs software that does not support ELS, see *Creating a Private VLAN on a Single EX Series Switch (CLI Procedure)*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

For security reasons, it is often useful to restrict the flow of broadcast and unknown unicast traffic or limit the communication between known hosts. Private VLANs (PVLANS) enable you to split a broadcast domain (primary VLAN) into multiple isolated broadcast subdomains (secondary VLANs), essentially putting a VLAN inside a VLAN. This procedure describes how to create a PVLAN on a single switch.



NOTE: You must specify a VLAN ID for each secondary VLAN even if the PVLAN is configured on a single switch.

You do not need to preconfigure the primary VLAN. This topic shows the primary VLAN being configured as part of this PVLAN configuration procedure.

For a list of guidelines on configuring PVLANS, see *Understanding Private VLANs*.

To configure a private VLAN on a single switch:

1. Set the VLAN ID for the primary VLAN:

[edit vlans]

```
user@switch# set primary-vlan-name vlan-id vlan-id-number
```

2. Configure at least one interface within the primary VLAN so that it communicates with all the subdomains of the PVLAN. This interface functions as a *promiscuous* port. It can be either a trunk port or an access port.

[edit interfaces]

```
user@switch# set interface-name unit 0 family ethernet-switching
```

```
user@switch# set interface-name unit 0 family ethernet-switching vlan members  
primary-vlan-name
```

3. Configure another promiscuous interface of the primary VLAN as a trunk port to connect the PVLAN to the external router or switch:

[edit interfaces]

```
user@switch# set interface-name unit 0 family ethernet-switching interface-mode trunk
```

```
user@switch# set interface-name unit 0 family ethernet-switching vlan members  
primary-vlan-name
```

4. Create an isolated VLAN by selecting the **isolated** option for **private-vlan**, and setting a VLAN ID for the isolated VLAN:

[edit vlans]

```
user@switch# set isolated-vlan-name private-vlan isolated vlan-id isolated-vlan-id
```



NOTE: You can create only one isolated VLAN within a private VLAN. Setting the VLAN name for the isolated VLAN is optional. Configuring the VLAN ID is required.

5. Create a community VLAN by selecting the **community** option for **private-vlan**, and setting a VLAN ID for this community VLAN:

[edit vlans]

```
user@switch# set community-vlan-name private-vlan community vlan-id community-vlan-id
```



NOTE: To create additional community VLANs, repeat this step and specify a different name for the community VLAN. Setting the VLAN name for the community VLAN is optional. Configuring the VLAN ID is required.

6. Associate the isolated VLAN with the primary VLAN:

[edit vlans]

```
user@switch# set primary-vlan-name vlan-id primary-vlan-id isolated-vlan isolated-vlan-name
```

7. Associate each community VLAN with the primary VLAN:

[edit vlans]

```
user@switch# set primary-vlan-name vlan-id primary-vlan-id
community-vlan community-vlan-name
```

8. If you have not already done so, configure at least one interface of the isolated VLAN.

[edit interfaces]

```
user@switch# set interface-name unit logical-unit-number family ethernet-switching
interface-mode access vlan members isolated-vlan-name
```

9. If you have not already done so, configure at least one interface of the community VLAN.

[edit interfaces]

```
user@switch# set interface-name unit logical-unit-number family ethernet-switching
interface-mode access vlan members community-vlan-name
```



NOTE: Repeat the same step on other community VLANs that you want to include in the PVLAN.

Related Documentation

- [Understanding Private VLANs](#)
- [Creating a Private VLAN Spanning Multiple Switches \(CLI Procedure\) on page 216](#)

Creating a Private VLAN Spanning Multiple Switches (CLI Procedure)



NOTE: This task uses Junos OS for EX Series switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Creating a Private VLAN Spanning Multiple EX Series Switches (CLI Procedure)*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

For security reasons, it is often useful to restrict the flow of broadcast and unknown unicast traffic or limit the communication between known hosts. Private VLANs (PVLANS) enable you to split a broadcast domain (primary VLAN) into multiple isolated broadcast subdomains (secondary VLANs), essentially putting a VLAN inside a VLAN. This procedure describes how to configure a PVLAN to span multiple switches.

For a list of guidelines on configuring PVLANS, see *Understanding Private VLANs*.

To configure a PVLAN to span multiple switches, perform the following procedure on all the switches that will participate in the PVLAN::

1. Create the primary VLAN by setting the unique VLAN name and specify an 802.1Q tag for the VLAN:

```
[edit vlans]
user@switch# set primary-vlan-name vlan-id number
```

2. On the switch that will connect to a router, configure a promiscuous interface as a trunk port to connect the PVLAN to the router:

```
[edit interfaces]
user@switch# set interface-name unit 0 family ethernet-switching interface-mode trunk
user@switch# set interface-name unit 0 family ethernet-switching vlan members
primary-vlan-name
```

3. On all the switches, configure a trunk interface as the Inter-Switch Link (ISL) that will be used to connect the switches to each other:

```
[edit interfaces]
user@switch# set interface-name unit 0 family ethernet-switching interface-mode trunk
inter-switch-link
user@switch# set interface-name unit 0 family ethernet-switching vlan members
name-of-private-vlan
```

4. Create an isolated VLAN within the primary VLAN by selecting the **isolated** option for **private-vlan**, and setting a VLAN ID for the isolated VLAN:

```
[edit vlans]
user@switch# set isolated-vlan-name private-vlan isolated vlan-id isolated-vlan-id
```



NOTE: You can create only one isolated VLAN within a private VLAN. The isolated VLAN can contain member interfaces from the multiple switches that compose the PVLAN.

Setting the VLAN name for the isolated VLAN is optional. Configuring the VLAN ID is required.

5. Create a community VLAN within the primary VLAN by selecting the **community** option for **private-vlan**, and setting a VLAN ID for this community VLAN::

[edit vlans]

```
user@switch# set community-vlan-name private-vlan community vlan-id community-vlan-id
```



NOTE: To create additional community VLANs, repeat this step and specify a different name for the community VLAN. Setting the VLAN name for the community VLAN is optional. Configuring the VLAN ID is required.

6. Associate the isolated VLAN with the primary VLAN:

[edit vlans *primary-vlan-name* vlan-id *primary-vlan-id*]

```
user@switch# set isolated-vlan isolated-vlan-name
```

7. Associate each community VLAN with the primary VLAN:

[edit vlans *primary-vlan-name* vlan-id *primary-vlan-id*]

```
user@switch# set community-vlan community-vlan-name
```

8. If you have not already done so, configure at least one access interface to be a member of the isolated VLAN.

[edit interface]

```
user@switch# set interface-name unit logical-unit-number family ethernet-switching
interface-mode access vlan members isolated-vlan-name
```

9. If you have not already done so, configure at least one access interface to be a member of the community VLAN.

[edit interface]

```
user@switch# set interface-name unit logical-unit-number family ethernet-switching
interface-mode access vlan members community-vlan-name
```



NOTE: Repeat this step for the other community VLANs that you are including in the PVLAN.

Related Documentation

- [Understanding Private VLANs](#)
- [Example: Configuring a Private VLAN on a Single Switch on page 217](#)
- [Creating a Private VLAN on a Single Switch \(CLI Procedure\) on page 214](#)

Example: Configuring a Private VLAN on a Single Switch



NOTE: This example uses Junos OS for switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your EX switch runs software that does not support ELS, see *Example: Configuring a Private VLAN on a Single EX Series Switch*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

For security reasons, it is often useful to restrict the flow of broadcast and unknown unicast traffic or limit the communication between known hosts. Private VLANs (PVLANS) enable you to split a broadcast domain (primary VLAN) into multiple isolated broadcast subdomains (secondary VLANs), essentially putting a VLAN inside a VLAN.

This example describes how to create a PVLAN on a single switch:

- [Requirements on page 218](#)
- [Overview and Topology on page 218](#)
- [Configuration on page 219](#)
- [Verification on page 221](#)

Requirements

This example uses the following hardware and software components:

- One Junos OS switch
- Junos OS Release 14.1X53-D10 or later for EX Series switches
Junos OS Release 14.1X53-D15 or later for QFX Series switches

Overview and Topology

You can isolate groups of subscribers for improved security and efficiency. This configuration example uses a simple topology to illustrate how to create a PVLAN with one primary VLAN and three secondary VLANs (one isolated VLAN, and two community VLANs).

[Table 22 on page 218](#) lists the interfaces of the topology used in the example.

Table 22: Interfaces of the Topology for Configuring a PVLAN

Interface	Description
ge-0/0/0 ge-1/0/0	Promiscuous member ports
ge-0/0/11, ge-0/0/12	HR community VLAN member ports
ge-0/0/13, ge-0/0/14	Finance community VLAN member ports
ge-0/0/15, ge-0/0/16	Isolated member ports

[Table 23 on page 218](#) lists the VLAN IDs of the topology used in the example.

Table 23: VLAN IDs in the Topology for Configuring a PVLAN

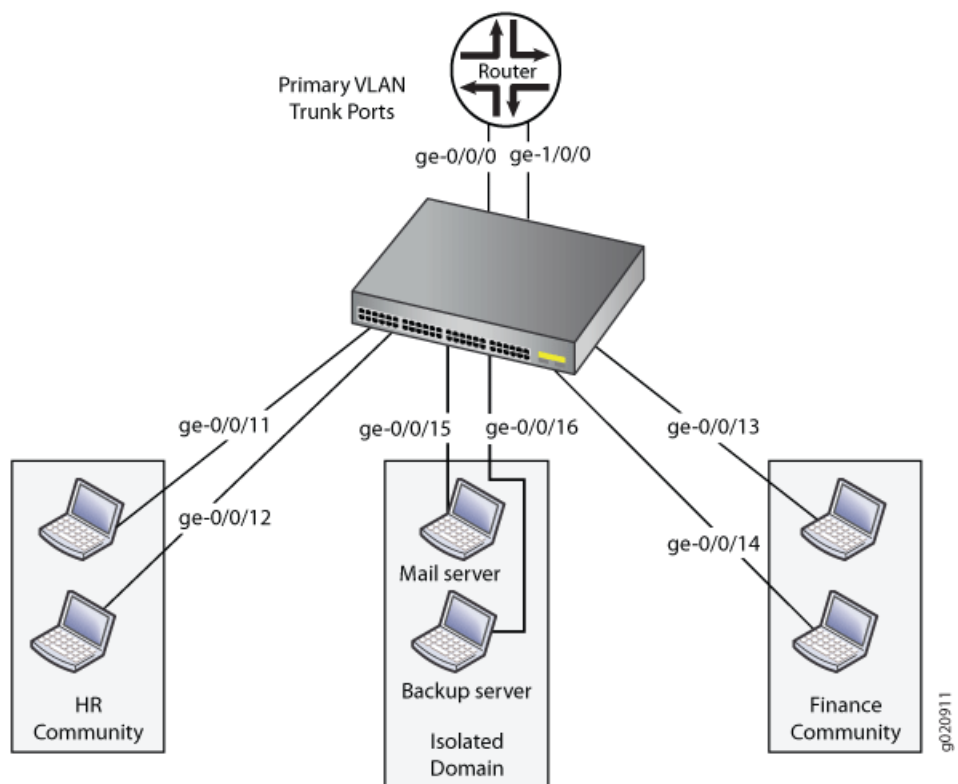
VLAN ID	Description
100	Primary VLAN
200	HR community VLAN

Table 23: VLAN IDs in the Topology for Configuring a PVLAN (*continued*)

VLAN ID	Description
300	Finance community VLAN
400	Isolated VLAN

Figure 16 on page 219 shows the topology for this example.

Figure 16: Topology of a Private VLAN on a Single EX Series Switch



Configuration

You can use an existing VLAN as the basis for your private PVLAN and create subdomains within it. This example creates a primary VLAN—using the VLAN name **vlan-pri**—as part of the procedure.

To configure a PVLAN, perform these tasks:

CLI Quick Configuration

To quickly create and configure a PVLAN, copy the following commands and paste them into the switch terminal window:

```
[edit]
set vlans vlan-pri vlan-id 100
set vlans vlan-iso private-vlan isolated vlan-id 400
set vlans vlan-hr private-vlan community vlan-id 200
set vlans vlan-finance private-vlan community vlan-id 300
```

```

set vlans vlan-pri vlan-id 100 isolated-vlan vlan-iso community-vlan vlan-hr community-vlan
vlan-finance
set interface ge-0/0/15 unit 0 family ethernet-switching interface-mode access vlan members
vlan-iso
set interface ge-0/0/16 unit 0 family ethernet-switching interface-mode access vlan members
vlan-iso
set interface ge-0/0/11 unit 0 family ethernet-switching interface-mode access vlan members
vlan-hr
set interface ge-0/0/12 unit 0 family ethernet-switching interface-mode trunk vlan members
vlan-hr
set interface ge-0/0/13 unit 0 family ethernet-switching interface-mode access vlan members
vlan-finance
set interface ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk vlan members
vlan-finance
set interface ge-0/0/0 unit 0 family ethernet-switching interface-mode trunk vlan members
vlan-pri
set interface ge-1/0/0 unit 0 family ethernet-switching interface-mode trunk vlan members
vlan-pri

```

Step-by-Step Procedure

To configure the PVLAN:

1. Create the primary VLAN (in this example, the name is **vlan-pri**) of the private VLAN:


```

[edit vlans]
user@switch# set vlan-pri vlan-id 100

```
2. Create an isolated VLAN and assign it a VLAN ID:


```

[edit vlans]
user@switch# set vlan-iso private-vlan isolated vlan-id 400

```
3. Create the HR community VLAN and assign it a VLAN ID:


```

[edit vlans]
user@switch# set vlan-hr private-vlan community vlan-id 200

```
4. Create the finance community VLAN and assign it a VLAN ID:


```

[edit vlans]
user@switch# set vlan-finance private-vlan community vlan-id 300

```
5. Associate the secondary VLANs with the primary VLAN:


```

[edit vlans]
user@switch# set vlan-pri vlan-id 100 isolated-vlan vlan-iso community-vlan vlan-hr
community-vlan vlan-finance

```
6. Set the interfaces to the appropriate interface modes:


```

[edit interfaces]
user@switch# set ge-0/0/15 unit 0 family ethernet-switching interface-mode access vlan
members vlan-iso
user@switch# set ge-0/0/16 unit 0 family ethernet-switching interface-mode trunk vlan
members vlan-iso
user@switch# set ge-0/0/11 unit 0 family ethernet-switching interface-mode access vlan
members vlan-hr
user@switch# set ge-0/0/12 unit 0 family ethernet-switching interface-mode access vlan
members vlan-hr
user@switch# set ge-0/0/13 unit 0 family ethernet-switching interface-mode access vlan
members vlan-finance
user@switch# set ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk vlan
members vlan-finance

```
7. Configure a promiscuous trunk interface of the primary VLAN. This interface is used by the primary VLAN to communicate with the secondary VLANs.

```
user@switch# set ge-0/0/0 unit 0 family ethernet-switching interface-mode trunk vlan
members vlan-pri
```

8. Configure another trunk interface (it is also a promiscuous interface) of the primary VLAN, connecting the PVLAN to the router.

```
user@switch# set ge-1/0/0 unit 0 family ethernet-switching interface-mode trunk vlan
members vlan-pri
```

Results

Check the results of the configuration:

```
user@switch> show configuration
```

Verification

To confirm that the configuration is working properly, perform these tasks:

- [Verifying That the Private VLAN and Secondary VLANs Were Created on page 221](#)

Verifying That the Private VLAN and Secondary VLANs Were Created

Purpose	Verify that the primary VLAN and secondary VLANs were properly created on the switch.
Action	Use the show vlans command: user@switch> show vlans extensive
Meaning	The output shows that the primary VLAN was created and identifies the interfaces and secondary VLANs associated with it.
Related Documentation	<ul style="list-style-type: none"> • Understanding Private VLANs • Creating a Private VLAN on a Single Switch (CLI Procedure) on page 214 • Creating a Private VLAN Spanning Multiple Switches (CLI Procedure) on page 216

Verifying That a Private VLAN Is Working

Purpose After creating and configuring private VLANs (PVLANS), verify that they are set up properly.

Action 1. To determine whether you successfully created the primary and secondary VLAN configurations:

- For a PVLAN on a single switch, use the **show configuration vlans** command:

```
user@switch> show configuration vlans
community1 {
    interface {
        interface a;
        interface b;
    }
    primary-vlan pvlan;
}
community2 {
    interface {
```

```

        interface d;
        interface e;
    }
    primary-vlan pvlan;
}
pvlan {
    vlan-id 1000;
    interface {
        isolated1;
        isolated2;
        trunk1;
        trunk2;
    }
    no-local-switching;
}

```

- For a PVLAN spanning multiple switches, use the **show vlans extensive** command:

```

user@switch> show vlans extensive
VLAN: COM1, Created at: Tue May 11 18:16:05 2010
802.1Q Tag: 100, Internal index: 3, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 1 (Active = 1)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/7.0*, untagged, access

```

```

VLAN: __pvlan_primary_ge-0/0/0.0__, Created at: Tue May 11 18:16:05 2010
Internal index: 5, Admin State: Enabled, Origin: Static
Private VLAN Mode: Isolated, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 1 (Active = 1)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/0.0*, untagged, access

```

```

VLAN: __pvlan_primary_ge-0/0/2.0__, Created at: Tue May 11 18:16:05 2010
Internal index: 6, Admin State: Enabled, Origin: Static
Private VLAN Mode: Isolated, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 1 (Active = 0)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/2.0, untagged, access

```

```

VLAN: __pvlan_primary_isiv__, Created at: Tue May 11 18:16:05 2010
802.1Q Tag: 50, Internal index: 7, Admin State: Enabled, Origin: Static
Private VLAN Mode: Inter-switch-isolated, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 0 (Active = 0)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk

```

```

VLAN: community2, Created at: Tue May 11 18:16:05 2010

```

```

802.1Q Tag: 20, Internal index: 8, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 2 (Active = 2)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/1.0*, untagged, access
    ge-1/0/6.0*, untagged, access

```

```

VLAN: primary, Created at: Tue May 11 18:16:05 2010
802.1Q Tag: 10, Internal index: 2, Admin State: Enabled, Origin: Static
Private VLAN Mode: Primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 5 (Active = 4)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/0.0*, untagged, access
    ge-0/0/1.0*, untagged, access
    ge-0/0/2.0, untagged, access
    ge-0/0/7.0*, untagged, access
    ge-1/0/6.0*, untagged, access

```

```

Secondary VLANs: Isolated 2, Community 2, Inter-switch-isolated 1
Isolated VLANs :
    __pvlan_primary_ge-0/0/0.0__
    __pvlan_primary_ge-0/0/2.0__
Community VLANs :
    COM1
    community2
Inter-switch-isolated VLAN :
    __pvlan_primary_isiv__

```

2. Use the **show vlans extensive** command to view VLAN information and link status for a PVLAN on a single switch or for a PVLAN spanning multiple switches.

- For a PVLAN on a single switch:

```

user@switch> show vlans pvlan extensive
VLAN: pvlan, Created at: time
802.1Q Tag: vlan-id, Internal index: index-number, Admin State: Enabled,
Origin: Static
Private VLAN Mode: Primary
Protocol: Port Mode
Number of interfaces: Tagged 2 (Active = 0), Untagged 6 (Active = 0)
    trunk1, tagged, trunk
    interface a, untagged, access
    interface b, untagged, access
    interface c, untagged, access
    interface d, untagged, access
    interface e, untagged, access
    interface f, untagged, access
    trunk2, tagged, trunk
Secondary VLANs: Isolated 2, Community 2
Isolated VLANs :
    __pvlan_pvlan_isolated1__
    __pvlan_pvlan_isolated2__
Community VLANs :

```

```
community1
community2
```

- For a PVLAN spanning multiple switches:

```
user@switch> show vlans extensive
```

```
VLAN: COM1, Created at: Tue May 11 18:16:05 2010
802.1Q Tag: 100, Internal index: 3, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 1 (Active = 1)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/7.0*, untagged, access
```

```
VLAN: __pvlan_primary_ge-0/0/0.0__, Created at: Tue May 11 18:16:05 2010
Internal index: 5, Admin State: Enabled, Origin: Static
Private VLAN Mode: Isolated, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 1 (Active = 1)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/0.0*, untagged, access
```

```
VLAN: __pvlan_primary_ge-0/0/2.0__, Created at: Tue May 11 18:16:05 2010
Internal index: 6, Admin State: Enabled, Origin: Static
Private VLAN Mode: Isolated, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 1 (Active = 0)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/2.0, untagged, access
```

```
VLAN: __pvlan_primary_isiv__, Created at: Tue May 11 18:16:05 2010
802.1Q Tag: 50, Internal index: 7, Admin State: Enabled, Origin: Static
Private VLAN Mode: Inter-switch-isolated, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 0 (Active = 0)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
```

```
VLAN: community2, Created at: Tue May 11 18:16:05 2010
802.1Q Tag: 20, Internal index: 8, Admin State: Enabled, Origin: Static
Private VLAN Mode: Community, Primary VLAN: primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 2 (Active = 2)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/1.0*, untagged, access
    ge-1/0/6.0*, untagged, access
```

```
VLAN: primary, Created at: Tue May 11 18:16:05 2010
802.1Q Tag: 10, Internal index: 2, Admin State: Enabled, Origin: Static
```

```

Private VLAN Mode: Primary
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 3 (Active = 3), Untagged 5 (Active = 4)
    ge-0/0/20.0*, tagged, trunk
    ge-0/0/22.0*, tagged, trunk, pvlan-trunk
    ge-0/0/23.0*, tagged, trunk, pvlan-trunk
    ge-0/0/0.0*, untagged, access
    ge-0/0/1.0*, untagged, access
    ge-0/0/2.0, untagged, access
    ge-0/0/7.0*, untagged, access
    ge-1/0/6.0*, untagged, access

Secondary VLANs: Isolated 2, Community 2, Inter-switch-isolated 1
Isolated VLANs :
    __pvlan_primary_ge-0/0/0.0__
    __pvlan_primary_ge-0/0/2.0__
Community VLANs :
    COM1
    community2
Inter-switch-isolated VLAN :
    __pvlan_primary_isiv__

```

3. Use the **show ethernet-switching table** command to view logs for MAC learning on the VLANs:

```

user@switch> show ethernet-switching table
Ethernet-switching table: 8 entries, 1 learned

```

VLAN	MAC address	Type	Age	Interfaces
default	*	Flood	-	All-members
pvlan	*	Flood	-	All-members
pvlan	MAC1	Replicated	-	interface a
pvlan	MAC2	Replicated	-	interface c
pvlan	MAC3	Replicated	-	isolated2
pvlan	MAC4	Learn	0	trunk1
__pvlan_pvlan_isolated1__ *		Flood	-	All-members
__pvlan_pvlan_isolated1__ MAC4		Replicated	-	trunk1
__pvlan_pvlan_isolated2__ *		Flood	-	All-members
__pvlan_pvlan_isolated2__ MAC3		Learn	0	isolated2
__pvlan_pvlan_isolated2__ MAC4		Replicated	-	trunk1
community1	*	Flood	-	All-members
community1	MAC1	Learn	0	interface a
community1	MAC4	Replicated	-	trunk1
community2	*	Flood	-	All-members

community2	MAC2	Learn	0 interface c
community2	MAC4	Replicated	- trunk1



NOTE: If you have configured a PVLAN spanning multiple switches, you can use the same command on all the switches to check the logs for MAC learning on those switches.

Meaning In the output displays for a PVLAN on a single switch, you can see that the primary VLAN contains two community domains (**community1** and **community2**), two isolated ports, and two trunk ports. The PVLAN on a single switch has only one tag (**1000**), which is for the primary VLAN.

The PVLAN that spans multiple switches contains multiple tags:

- The community domain **COM1** is identified with tag **100**.
- The community domain **community2** is identified with tag **20**.
- The interswitch isolated domain is identified with tag **50**.
- The primary VLAN **primary** is identified with tag **10**.

Also, for the PVLAN that spans multiple switches, the trunk interfaces are identified as **pvlan-trunk**.

- Related Documentation**
- *Creating a Private VLAN on a Single EX Series Switch (CLI Procedure)*
 - *Creating a Private VLAN Spanning Multiple EX Series Switches (CLI Procedure)*
 - *Creating a Private VLAN on a Single Switch*
 - *Creating a Private VLAN Spanning Multiple Switches*

PART 7

Proxy ARP

- [Using Proxy ARP on page 229](#)

CHAPTER 7

Using Proxy ARP

- [Understanding Proxy ARP on page 229](#)
- [Configuring Proxy ARP \(CLI Procedure\) on page 231](#)
- [Verifying That Proxy ARP Is Working Correctly on page 231](#)

Understanding Proxy ARP

You can configure proxy Address Resolution Protocol (ARP) to enable the switch to respond to ARP queries for network addresses by offering its own Ethernet media access control (MAC) address. With proxy ARP enabled, the switch captures and routes traffic to the intended destination.

Proxy ARP is useful in situations where hosts are on different physical networks and you do not want to use subnet masking. Because ARP broadcasts are not propagated between hosts on different physical networks, hosts will not receive a response to their ARP request if the destination is on a different subnet. Enabling the switch to act as an ARP proxy allows the hosts to transparently communicate with each other through the switch. Proxy ARP can help hosts on a subnet reach remote subnets without your having to configure routing or a default gateway.

- [What Is ARP? on page 229](#)
- [Proxy ARP Overview on page 229](#)
- [Best Practices for Proxy ARP on page 230](#)

What Is ARP?

Ethernet LANs use ARP to map Ethernet MAC addresses to IP addresses. Each device maintains a cache containing a mapping of MAC addresses to IP addresses. The switch maintains this mapping in a cache that it consults when forwarding packets to network devices. If the ARP cache does not contain an entry for the destination device, the host (the DHCP client) broadcasts an ARP request for that device's address and stores the response in the cache.

Proxy ARP Overview

When proxy ARP is enabled, if the switch receives an ARP request for which it has a route to the target (destination) IP address, the switch responds by sending a proxy ARP reply

packet containing its own MAC address. The host that sent the ARP request then sends its packets to the switch, which forwards them to the intended host.



NOTE: For security reasons, the source address in an ARP request must be on the same subnet as the interface on which the ARP request is received.

You can configure proxy ARP for each interface. You can also configure proxy ARP for a VLAN by using a routed VLAN interface (RVI).

Two modes of proxy ARP are supported: restricted and unrestricted. Both modes require that the switch have an active route to the destination address of the ARP request.

- **Restricted**—The switch responds to ARP requests in which the physical networks of the source and target are different and does not respond if the source and target IP addresses are on the same subnet. In this mode, hosts on the same subnet communicate without proxy ARP. We recommend that you use this mode on the switch.
- **Unrestricted**—The switch responds to all ARP requests for which it has a route to the destination. This is the default mode (because it is the default mode in Juniper Networks Junos operating system (Junos OS) configurations other than those on the switch). We recommend using restricted mode on the switch.

Best Practices for Proxy ARP

We recommend these best practices for configuring proxy ARP on the switches:

- Set proxy ARP to restricted mode.
- Use restricted mode when configuring proxy ARP on RVIs.
- If you set proxy ARP to unrestricted, disable gratuitous ARP requests on each interface enabled for proxy ARP.

- Related Documentation**
- *Configuring Proxy ARP*
 - *proxy-arp*

Configuring Proxy ARP (CLI Procedure)



NOTE: This task uses Junos OS for EX Series switches and QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Configuring Proxy ARP (CLI Procedure)* or *Configuring Proxy ARP*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

You can configure proxy Address Resolution Protocol (ARP) on your switch to enable the switch to respond to ARP queries for network addresses by offering its own media access control (MAC) address. With proxy ARP enabled, the switch captures and routes traffic to the intended destination.

To configure proxy ARP on a single interface:

```
[edit interfaces]
user@switch# set interface-name unit logical-unit-number proxy-arp (restricted |
unrestricted)
```



BEST PRACTICE: We recommend that you configure proxy ARP in restricted mode. In restricted mode, the switch does not act as a proxy if the source and target IP addresses are on the same subnet. If you decide to use unrestricted mode, disable gratuitous ARP requests on the interface to avoid a situation wherein the switch's response to a gratuitous ARP request appears to the host to be an indication of an IP conflict.

To configure proxy ARP on an integrated routing and bridging (IRB) interface:

```
[edit interfaces]
user@switch# set irb.logical-unit-number proxy-arp restricted
```

- Related Documentation**
- [Example: Configuring Proxy ARP on an EX Series Switch](#)
 - [Verifying That Proxy ARP Is Working Correctly on page 231](#)
 - [Configuring Integrated Routing and Bridging Interfaces \(CLI Procedure\)](#)

Verifying That Proxy ARP Is Working Correctly

Purpose Verify that the switch is sending proxy ARP messages.

Action List the system statistics for ARP:

```
user@switch> show system statistics arp
arp:
    90060 datagrams received
    34 ARP requests received
    610 ARP replies received
    2 resolution request received
    0 unrestricted proxy requests
    0 restricted proxy requests
```

```
0 received proxy requests
0 unrestricted proxy requests not proxied
0 restricted proxy requests not proxied
0 datagrams with bogus interface
0 datagrams with incorrect length
0 datagrams for non-IP protocol
0 datagrams with unsupported op code
0 datagrams with bad protocol address length
0 datagrams with bad hardware address length
0 datagrams with multicast source address
0 datagrams with multicast target address
0 datagrams with my own hardware address
0 datagrams for an address not on the interface
0 datagrams with a broadcast source address
294 datagrams with source address duplicate to mine
89113 datagrams which were not for me
0 packets discarded waiting for resolution
0 packets sent after waiting for resolution
309 ARP requests sent
35 ARP replies sent
0 requests for memory denied
0 requests dropped on entry
0 requests dropped during retry
0 requests dropped due to interface deletion
0 requests on unnumbered interfaces
0 new requests on unnumbered interfaces
0 replies for from unnumbered interfaces
0 requests on unnumbered interface with non-subnetted donor
0 replies from unnumbered interface with non-subnetted donor
```

Meaning The statistics show that two proxy ARP requests were received. The **unrestricted proxy requests not proxied** and **restricted proxy requests not proxied** fields indicate that all the unproxied ARP requests received have been proxied by the switch.

Related Documentation

- [Configuring Proxy ARP](#)
- [Configuring Proxy ARP \(CLI Procedure\) on page 231](#)

PART 8

Reflective Relay

- [Using Reflective Relay on page 235](#)

CHAPTER 8

Using Reflective Relay

- [Understanding Reflective Relay for Use with VEPA Technology on page 235](#)
- [Configuring Reflective Relay on page 236](#)
- [Example: Configuring Reflective Relay for Use with VEPA Technology on page 237](#)

Understanding Reflective Relay for Use with VEPA Technology

Virtual Ethernet Port Aggregator (VEPA) technology aggregates packets generated by virtual machines located on the same server and relays them to a physical switch. The physical switch then provides connectivity between the virtual machines located on the server, so the virtual machines do not communicate with one another. Offloading switching activities from a virtual switch to a physical switch reduces the computing overhead on the virtual servers and takes advantage of the security, filtering, and management features of the physical switch. Reflective relay, also known as “hairpin turn,” enables the physical switch to receive aggregated packets from the virtual machines hosted on the server through the VEPA on the downstream port and send those packets out the same downstream port from which the physical switch received them.

- [VEPA on page 235](#)
- [Reflective Relay on page 235](#)

VEPA

Even though virtual machines are capable of sending packets directly to one another, it is more efficient to pass these aggregated packets from the VEPA to a physical switch. The switch can then send any packets destined for a virtual machine located on the same server to the VEPA.

Reflective Relay

Reflective relay, also known as a “hairpin turn” or “hairpin mode,” returns aggregated packets to the VEPA by using the same downstream port that initially delivered the aggregated packets from the VEPA to the switch. Reflective relay must be configured on the interface located on the physical switch that receives aggregated packets, such as VEPA packets, because some of these packets might need to be sent back to the server if they are destined for another virtual machine on the same server.

Reflective relay only occurs in two situations:

- When the destination address of the packet was learned on that downstream port
- When the destination has not yet been learned

Reflective relay does not otherwise change the operation of the switch. If the interface to which the virtual machine is connected and the MAC address of the virtual machine packet are not yet included in the Ethernet switching table for the virtual machine's associated VLAN, an entry is added. If the source MAC address of an incoming packet under the respective VLAN is not yet present in the Ethernet switching table, the switch floods the packet on all the other ports that are members of the same VLAN, including the port on which the packet arrived.

Related Documentation

- [Understanding Bridging and VLANs](#)
- [Understanding Bridging and VLANs on page 35](#)
- [Example: Configuring Reflective Relay for Use with VEPA Technology](#)

Configuring Reflective Relay

Configure reflective relay when a switch port must return packets on a downstream port. For example, configure reflective relay when a switch port receives aggregated virtual machine packets from a technology such as virtual Ethernet port aggregator (VEPA). When these packets are passed through the switch, reflective relay allows the switch to send those packets back on the same interface that was used for delivery.



NOTE: This task uses Junos OS for QFX3500 and QFX3600 switches that supports the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see [Configuring Reflective Relay](#).

Before you begin configuring reflective relay, ensure that you have:

- Configured packet aggregation on the server connected to the port. See your server documentation.
- Configured the port for all VLANs that could be included in aggregated packets..

To configure reflective relay:

1. Configure an Ethernet interface with an interface mode of **trunk**:

```
[edit]
user@switch# set interfaces interface-name unit number family family-type interface-mode
trunk
```

For example:

```
[edit]
user@switch# set interfaces xe-0/0/2 unit 0 family ethernet-switching interface-mode trunk
```

2. Configure the interface for reflective relay:

```
[edit]
user@switch# set interfaces interface-name unit number family family-type reflective-relay
```

For example:

```
[edit]
user@switch# set interfaces xe-0/0/2 unit 0 family ethernet-switching reflective-relay
```

3. Configure the interface for the VLANs that exist on the VM server:

```
[edit]
user@switch# set interfaces interface-name unit number family family-type vlan members
vlan-names
```

For example:

```
[edit]
user@switch# set interfaces xe-0/0/2 unit 0 family ethernet-switching vlan members
[VLAN_Purple VLAN_Orange VLAN_Blue]
```

Related Documentation

- [Example: Configuring Reflective Relay for Use with VEPA Technology on page 237](#)
- [Understanding Reflective Relay for Use with VEPA Technology on page 235](#)

Example: Configuring Reflective Relay for Use with VEPA Technology

Reflective relay must be configured on a switch that receives virtual machine aggregated packets, such as Virtual Ethernet Port Aggregator (VEPA) packets, because some of these packets might be sent back to the server destined for another virtual machine on the same server. Reflective relay returns those packets to the original device using the same downstream port that delivered the packets to the switch.



NOTE: This example uses Junos OS for QFX3500 and QFX3600 switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see *Example: Configuring Reflective Relay for Use with VEPA Technology*. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

This example shows how to configure a switch port interface to return packets sent by VEPA on the downstream interface back to the server using the same downstream interface:

- [Requirements on page 238](#)
- [Overview and Topology on page 238](#)
- [Configuration on page 240](#)
- [Verification on page 240](#)

Requirements

This example uses the following hardware and software components:

- One QFX3500 switch
- One server
- Junos OS Release 12.1 or later for the QFX Series

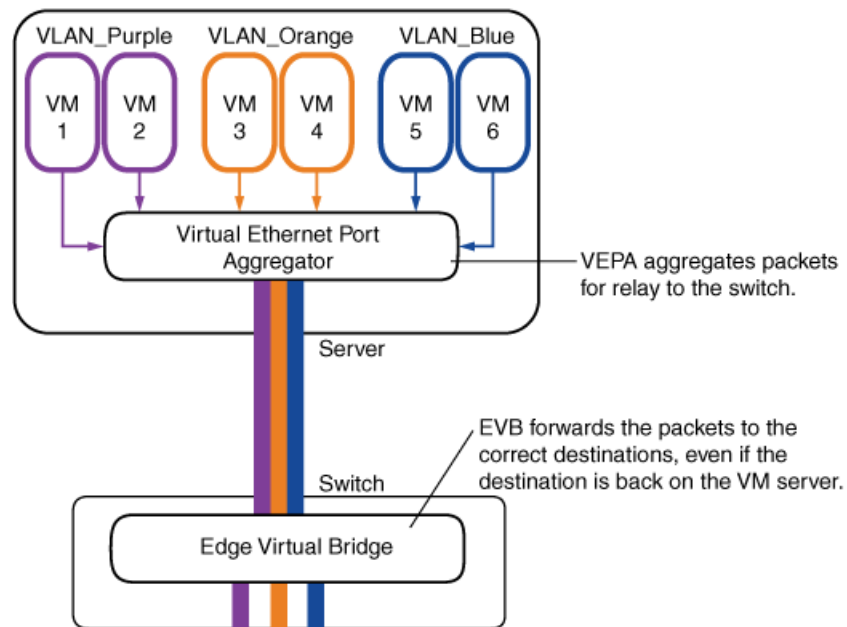
Before you configure reflective relay on a switch port, be sure you have:

- Configured a server with six virtual machines, VM 1 through VM 6.
- Configured the server with three VLANs named VLAN_Purple, VLAN_Orange, and VLAN_Blue and added two virtual machines to each VLAN.
- Configured the same three VLANs named VLAN_Purple, VLAN_Orange, and VLAN_Blue on one interface.
- Installed and configured VEPA to aggregate the virtual machine packets.

Overview and Topology

In this example, illustrated in [Figure 17 on page 239](#), a switch is connected to one server that is hosting six virtual machines and is configured with a VEPA for aggregating packets. The server's six virtual machines are VM1 through VM 6, and each virtual machine belongs to one of the three server VLANs, VLAN_Purple, VLAN_Orange, or VLAN_Blue. Instead of the server directly passing packets between virtual machines, packets from any of the three VLANs that are destined for another one of the three VLANs are aggregated using VEPA technology and passed to the switch for processing. You must configure the switch port to accept these aggregated packets on the downstream interface and to return appropriate packets to the server on the same downstream interface after they are processed. [Figure 17 on page 239](#) shows the topology for this example.

Figure 17: Reflective Relay Topology



g020996

In this example, you configure the physical Ethernet switch port interface for trunk interface mode and reflective relay. Configuring trunk port mode allows the interface to accept VLAN tagged packets. Configuring reflective relay allows the downstream port to return those packets on the same interface. [Table 24 on page 239](#) shows the components used in this example.

Table 24: Components of the Topology for Configuring Reflective Relay

Component	Description
QFX3500 switch	Switch that supports reflective relay. For a list of switches that support this feature, see <i>QFX Series Software Features Overview</i> .
xe-0/0/2	Switch interface to the server.
Server	Server with virtual machines and VEPA technology.
Virtual machines	Six virtual machines located on the server: V1, V2, V3, V4, V5, and V6.
VLANs	Three VLANs: VLAN_Purple, VLAN_Orange, and VLAN_Blue. Each VLAN has two virtual machine members.
VEPA	Virtual Ethernet port aggregator that aggregates virtual machine packets on the server before the resulting single stream is transmitted to the switch.

Configuration

To configure reflective relay, perform these tasks:

- [Configuring Reflective Relay on the Port on page 240](#)

Configuring Reflective Relay on the Port

CLI Quick Configuration

To quickly configure reflective relay, copy the following commands and paste them into the switch window:

```
[edit]
set interfaces xe-0/0/2 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/2 unit 0 family ethernet-switching reflective-relay
set interfaces xe-0/0/2 unit 0 family ethernet-switching vlan members [VLAN_Blue VLAN_Orange
VLAN_Purple]
```

Step-by-Step Procedure

To configure reflective relay:

1. Configure the trunk interface mode on the interface:

```
[edit]
user@switch# set interfaces xe-0/0/2 unit 0 family ethernet-switching interface-mode
trunk
```

2. Configure reflective relay on the interface to allow it to both accept and send packets:

```
[edit]
user@switch# set interfaces xe-0/0/2 unit 0 family ethernet-switching reflective-relay
```

3. Configure the interface for the three VLANs on the server:

```
[edit]
user@switch# set interfaces xe-0/0/2 unit 0 family ethernet-switching vlan members
[VLAN_Purple VLAN_Orange VLAN_Blue]
```

Results

Check the results of the configuration:

```
[edit interfaces xe-0/0/2]
user@switch# show
unit 0 {
    family ethernet-switching {
        interface-mode trunk;
        reflective-relay;
        vlan {
            members [ VLAN_Purple VLAN_Orange VLAN_Blue ];
        }
    }
}
```

Verification

To confirm that reflective relay is enabled and working correctly, perform these tasks:

- [Verifying That Reflective Relay Is Enabled and Working Correctly on page 240](#)

Verifying That Reflective Relay Is Enabled and Working Correctly

Purpose

Verify that reflective relay is enabled and working correctly.

Action Use the `show ethernet-switching interfaces detail` command to display the reflective relay status:

```
user@switch> show ethernet-switching interfaces xe-0/0/2 detail
Interface: xe-0/0/2, Index: 66, State: down, Interface mode: Trunk
Reflective Relay Status: Enabled
Ether type for the interface: 0x8100
VLAN membership:
  VLAN_Purple, 802.1Q Tag: 450, tagged, unblocked
  VLAN_Orange, 802.1Q Tag: 460, tagged, unblocked
  VLAN_Blue, 802.1Q Tag: 470, tagged, unblocked
Number of MACs learned on IFL: 0
```

Confirm that reflective relay is working by sending a Layer 2 broadcast message from one virtual machine to another virtual machine located on the same VLAN. Check the switch to verify that the switch sends the packets back on the same interface on which they were received. One way to check this is to set up port mirroring on the switch interface, connect a traffic generator to the mirrored interface, and use the traffic generator to examine packets.

Alternatively, if you do not have a traffic generator available, you can send traffic between two virtual machines with FTP, Telnet, or SSH, while running the `tcpdump` utility on the receiver virtual machine port to capture reflected packets.

Meaning The reflective relay status is **Enabled**, meaning that interface **xe-0/0/2** is configured for the trunk interface mode, which accepts VLAN-tagged packets, and for reflective relay, which accepts and returns packets on the same interface.

When the traffic generator shows packets arriving at the switch and returning to the server on the same interface, reflective relay is working.

Related Documentation

- [Understanding Reflective Relay for Use with VEPA Technology on page 235](#)
- [Configuring Port Mirroring](#)
- [interface-mode on page 289](#)
- [reflective-relay on page 371](#)

PART 9

Unified Forwarding Table

- [Using the Unified Forwarding Table on page 245](#)

CHAPTER 9

Using the Unified Forwarding Table

- Understanding the Unified Forwarding Table on page 245
- Configuring the Unified Forwarding Table on page 248

Understanding the Unified Forwarding Table

- Using the Unified Forwarding Table to Optimize Address Storage on page 245
- MAC Address and Host Address Memory Allocation on page 245
- Using the LPM Table with Junos OS 13.2X51-D10 on page 247
- LPM Table Memory Allocation on page 247

Using the Unified Forwarding Table to Optimize Address Storage

On QFX5100, EX4600, and OCX1100 switches, you can control the allocation of forwarding table memory available to store the following:

- MAC addresses.
- Layer 3 host entries.
- Longest prefix match (LPM) table entries.



NOTE: Starting with Junos OS 13.2X51-D15, you can allocate more memory to store prefixes in the range /65 to /127 range.

This feature gives you the flexibility to configure your switch to match the needs of your particular network environment. For example, you might configure the switch to store more MAC addresses in a Layer 2 network, such as a virtualized network with many servers and virtualized machines. On the other hand, if your switch is located in the routing core of a network or participates in an IP fabric, you probably want to maximize the number of routing table entries it can store. In this case, you would configure it to use the **lpm-profile**, which provides the most longest prefix match table entries.

MAC Address and Host Address Memory Allocation

There are several profiles that allocate memory differently for MAC addresses and host addresses. You configure the mix that best meets your needs by choosing the appropriate

profile. [Table 25 on page 246](#) lists the profiles you can choose and the associated maximum values for the MAC address and host table entries.

Table 25: Unified Forwarding Table Profiles

Profile Name	MAC Table	Host Table (unicast and multicast addresses)					
	MAC Addresses	IPv4 unicast	IPv6 unicast	IPv4 (*, G)	IPv4 (S, G)	IPv6 (*, G)	IPv6 (S, G)
l2-profile-one	288K	16K	8K	8K	8K	4K	4K
l2-profile-two	224K	80K	40K	40K	40K	20K	20K
l2-profile-three (default)	160K	144K	72K	72K	72K	36K	36K
l3-profile	96K	208K	104K	104K	104K	52K	52K
lpm-profile	32K	16K	8K	8K	8K	4K	4K
lpm-profile with unicast-in-lpm option	32K	(stored in LPM table)	(stored in LPM table)	8K	8K	4K	4K



NOTE: On QFX5100, EX4600, and OCX1100 switches, IPv4 and IPv6 host routes with ECMP next hops are stored in the host table. On QFX3500 and QFX3600 switches, these routes are stored in the LPM table.

Note that all entries in the host table share the same memory space. If the host table stores the maximum number of entries for any given type, the entire shared table is full and is unable to accommodate *any* entries of any other type. As you can see, different entry types occupy different amounts of memory. For example, an IPv6 unicast address occupies twice as much memory as an IPv4 unicast address, and an IPv6 multicast address occupies four times as much memory as an IPv4 unicast address.

[Table 26 on page 246](#) lists various valid combinations that the host table can store if you use the **l2-profile-one** profile. Each row in the table represents a case in which the host table is full and cannot accommodate any more entries. .

Table 26: Example Host Table Combinations Using l2-profile-one

IPv4 unicast	IPv6 unicast	IPv4 multicast (*, G)	IPv4 multicast (S, G)	IPv6 multicast (*, G)	IPv6 multicast (S, G)
16K	0	0	0	0	0
12K	2K	0	0	0	0
12K	0	2K	2K	0	0
8K	4K	0	0	0	0

Table 26: Example Host Table Combinations Using l2-profile-one (*continued*)

IPv4 unicast	IPv6 unicast	IPv4 multicast (*, G)	IPv4 multicast (S, G)	IPv6 multicast (*, G)	IPv6 multicast (S, G)
4K	2K	2K	2K	0	0
0	4K	0	0	1K	1K

Using the LPM Table with Junos OS 13.2X51-D10

The LPM table is also shared and the same principles apply. [Table 27 on page 247](#) provides examples of valid combinations that the LPM table can store, also using the **l2-profile-one** profile. Once again, each row in the table represents a case in which the table is full and cannot accommodate any more entries.

Table 27: Example LPM Table Combinations Using l2-profile-one Using Junos OS 13.2X51-D10

IPv4 entries	IPv6 Entries (prefix <= 64)	IPv6 Entries (prefix >= 65)
16K	0	0
0	8K	0
8K	4K	0
4K	4K	1K
4K	2K	2K



NOTE: If you want to use more than 16 IPv6 addresses with prefix lengths greater than 64 with Junos OS 13.2X51-D10, you must follow the instructions at [“Configuring the Unified Forwarding Table” on page 248](#). As that topic explains, if you increase the number of addresses with prefix lengths greater than 64, you reduce the amount of LPM-table memory available to store IPv6 addresses with prefixes less than or equal to 64.

LPM Table Memory Allocation

You configure the memory allocation for LPM table entries differently depending on which version of Junos OS you use. To learn how to configure memory allocation for LPM table entries see [“Configuring the Unified Forwarding Table” on page 248](#). Note that starting with Junos OS 14.1X53-D30 you can free memory in the host table by using the **unicast-in-lpm** option with the **lpm-profile** to store IPv4 and IPv6 unicast addresses in the LPM table instead of the host table. See [“Configuring the lpm-profile With Junos OS 14.1x53-D30 and Later” on page 253](#).

Release History Table

Release	Description
14.1X53-D30	Note that starting with Junos OS 14.1X53-D30 you can free memory in the host table by using the unicast-in-lpm option with the lpm-profile to store IPv4 and IPv6 unicast addresses in the LPM table instead of the host table.
13.2X51-D15	Starting with Junos OS 13.2X51-D15, you can allocate more memory to store prefixes in the range /65 to /127 range.

Related Documentation

- [Configuring the Unified Forwarding Table on page 248](#)

Configuring the Unified Forwarding Table

To optimize the way your switch allocates memory for different types of addresses, you can choose a unified forwarding table profile. In addition to choosing this profile, you can also decide how you want memory allocated for longest prefix match (LPM) entries.

- [Configuring an Address-Storage Profile on page 248](#)
- [Configuring the LPM Allocation on page 249](#)

Configuring an Address-Storage Profile

On QFX5100, EX4600, and OCX1100 switches, you can control the allocation of memory available to store the following:

- MAC addresses
- Layer 3 host entries
- Longest prefix match (LPM) table entries

You configure the mix that best meets your needs by choosing the appropriate profile. [Table 28 on page 248](#) lists the profiles you can choose and the maximum values for the MAC address and host table entries.

Table 28: Unified Forwarding Table Profiles

Profile Name	MAC Table	Host Table (unicast and multicast addresses)					
	MAC Addresses	IPv4 unicast	IPv6 unicast	IPv4 (*, G)	IPv4 (S, G)	IPv6 (*, G)	IPv6 (S, G)
l2-profile-one	288K	16K	8K	8K	8K	4K	4K
l2-profile-two	224K	80K	40K	40K	40K	20K	20K
l2-profile-three (default)	160K	144K	72K	72K	72K	36K	36K
l3-profile	96K	208K	104K	104K	104K	52K	52K

Table 28: Unified Forwarding Table Profiles (*continued*)

Profile Name	MAC Table	Host Table (unicast and multicast addresses)					
lpm-profile	32K	16K	8K	8K	8K	4K	4K

Note that if the host table stores the maximum number of entries for any given type, the entire table is full and is unable to accommodate *any* entries of any other type. For example, an IPv6 unicast address occupies twice as much memory as an IPv4 unicast address, and an IPv6 multicast address occupies four times as much memory as an IPv4 unicast address. For more information about valid combinations of table entries see [“Understanding the Unified Forwarding Table” on page 245](#).

To configure the profile that you want, enter and commit the following statement:

```
[edit]
user@switch# set chassis forwarding-options profile-name
```



NOTE: When you configure and commit a profile, the PFE process restarts and all the data interfaces on the switch go down and come back up.

However, starting with Junos OS Release 14.1X53-D40, Packet Forwarding Engines on switches in a Virtual Chassis or Virtual Chassis Fabric (VCF) do not automatically restart upon configuring a unified forwarding table profile change, to avoid Virtual Chassis or VCF instability after the change propagates to member switches and multiple Packet Forwarding Engines automatically restart at the same time. Instead, a message is displayed at the CLI prompt and logged to the switch's system log to notify you that the profile change does not take effect until the next time you reboot the Virtual Chassis or VCF. We recommend that you plan to make profile changes only when you can perform a Virtual Chassis or VCF system reboot immediately after committing the configuration update. Otherwise, the Virtual Chassis or VCF could become inconsistent if one or more members have a problem and restart with the new configuration before a planned system reboot activates the change on all members.

The settings for **l2-profile-three** are configured by default. That is, if you do not enter a **set forwarding-options chassis profile-name** statement, these settings are configured.

Configuring the LPM Allocation

In addition to choosing a profile, you can further optimize memory allocation for LPM table entries by configuring how many IPv6 prefixes in the range /65 through /127 you want the switch to store. The switch uses LPM entries during address lookup to match addresses to the most-specific (longest) applicable prefix. The procedures for configuring the LPM table are different depending on which version of Junos OS you are using.

- [Configuring the LPM Table With Junos OS 13.2X51-D10 and 13.2X52-D10 on page 250](#)
- [Configuring the LPM Table With Junos OS 13.2x51-D15 and Later on page 251](#)

Configuring the LPM Table With Junos OS 13.2X51-D10 and 13.2X52-D10

With Junos OS 13.2x51-D10 and 13.2X52-D10, the switch allocates memory for 16 IPv6 prefixes in the range /65 through /127 by default. If you want to use more than 16 IPv6 prefixes in this range, you must enter and commit the following statement:

```
user@switch# set chassis forwarding-options profile-name num-65-127-prefix [1-128]
```

Each increment adds support for 16 IPv6 prefixes between /65 and /127, for a maximum of 2048 such prefixes (16 x 128 = 2048). The system supports 16 of these prefixes by default, so to increase the number of supported prefixes, you must enter a value of 2 or greater. For example, if you enter **2**, the system will support 32 IPv6 prefixes in the range /65 through /127.



NOTE: When you configure and commit the `num-65-127-prefix` value, all the data interfaces on the switch restart. The management interfaces are unaffected.

The LPM table is shared, and each increment that you add for IPv6 prefixes in the range /65 through /127 reduces the number of table entries that are available for IPv4 prefixes and IPv6 prefixes shorter than /65. Note that IPv6 prefixes /65 and longer consume twice as much memory as shorter IPv6 prefixes and four times as much memory as IPv4 prefixes. So, for example, entering the following statement

```
user@switch# set chassis forwarding-options l2-profile-one num-65-127-prefix 2
```

provides for 16 additional IPv6 prefixes /65 or longer (for a total of 32 such prefixes) and reduces the numbers of other prefixes that can be stored, as indicated:

- 32 fewer IPv6 prefixes shorter than /65 (16 IPv6 prefixes /65 or longer consume the same amount of memory as 32 IPv6 prefixes shorter than /65), or
- 64 fewer IPv4 prefixes (16 IPv6 prefixes /65 or longer consume the same amount of memory as 64 IPv4 prefixes)

Table 27 on page 247 provides examples of valid combinations that the LPM table can store using the **l2** and **l3** profiles. Once again, each row in the table represents a case in which the table is full and cannot accommodate any more entries.

Table 29: Example LPM Table Combinations Using l2-and l3 Profiles With Junos OS 13.2X51-D10 and 13.2X52-D10

num-65-127-prefix Value	IPv4 Entries	IPv6 Entries (Prefix <= 64)	IPv6 Entries (Prefix >= 65)
1 (default)	16K-16	OK	16
1 (default)	OK	8K-16	16
1 (default)	8K-16	4K	16
64	4K	4K	1K

Table 29: Example LPM Table Combinations Using l2-and l3 Profiles With Junos OS 13.2X51-D10 and 13.2X52-D10 (*continued*)

num-65-127-prefix Value	IPv4 Entries	IPv6 Entries (Prefix <= 64)	IPv6 Entries (Prefix >= 65)
64	2K	5K	1K
64	0K	6K	1K
128	4K	2K	2K
128	2K	3K	2K
128	0K	4K	2K



NOTE: With Junos OS 13.2X51-D10 and 13.2X52-D10, the `lpm-profile` does not support IPv6 prefixes. If you use this version of Junos OS and also use the `lpm-profile`, do not configure the `num-65-127-prefix` statement. That is, leave it at its default value of 1, which allows for as many as 128K IPv4 prefixes (the maximum possible).

Configuring the LPM Table With Junos OS 13.2x51-D15 and Later

With Junos OS 13.2X51-D15 and later, you can configure the memory allocation for the LPM table for the `lpm-profile` profile independently of the other profiles. In addition, Junos OS 13.2x51-D15 offers twice as much storage for IPv6 prefixes /65 through /127 (4K instead of 2K) for the l2 and l3 profiles.

- [Configuring the l2 and l3 profiles With Junos OS 13.2x51-D15 and Later on page 251](#)
- [Configuring the lpm-profile With Junos OS 13.2x51-D15 and Later on page 252](#)
- [Configuring the lpm-profile With Junos OS 14.1x53-D30 and Later on page 253](#)

Configuring the l2 and l3 profiles With Junos OS 13.2x51-D15 and Later

With Junos OS 13.2x51-D15 and later, you can configure the switch to support as many as 4K IPv6 prefixes /65 through /127 if you are using any profile other than the `lpm-profile` profile. To do so, enter and commit the following statement:

```
user@switch# set chassis forwarding-options lpm-profile num-65-127-prefix [0-4]
```

Each increment adds support for 1K IPv6 prefixes between /65 and /127, for a maximum of 4K such prefixes. The default value is 1, which allocates memory for 1K of IPv6 prefixes in this range. Each increment that you add for IPv6 prefixes in the range /65 through /127 reduces the number of table entries that are available for IPv6 prefixes shorter than /65 and IPv4 prefixes. [Table 30 on page 252](#) shows the numbers of entries that you can allocate by using the `num-65-127-prefix` statement with Junos OS 13.2X51-D15. Once again, each row represents a case in which the table is full and cannot accommodate any more entries.

Table 30: LPM Table Combinations for l2 and l3 profiles With Junos OS 13.2X51-D15

num-65-127-prefix Value	IPv4 Entries	IPv6 Entries (Prefix <= 64)	IPv6 Entries (Prefix >= 65)
0	16K	8K	0K
1 (default)	12K	6K	1K
2	8K	4K	2K
3	4K	2K	3K
4	0K	0K	4K



NOTE: When you configure the `num-65-127-prefix` value, the PFE process restarts and all the data interfaces on the switch go down and come back up. The management interfaces are unaffected.

However, starting with Junos OS Release 14.1X53-D40, Packet Forwarding Engines on switches in a Virtual Chassis or Virtual Chassis Fabric (VCF) do not automatically restart upon configuring a unified forwarding table profile change, to avoid Virtual Chassis or VCF instability when the change propagates to member switches and multiple Packet Forwarding Engines restart at the same time. Instead, a message is displayed at the CLI prompt and logged to the switch's system log to notify you that the profile change does not take effect until the next time you reboot the Virtual Chassis or VCF. We recommend that you plan to make profile changes only when you can perform a Virtual Chassis or VCF system reboot immediately after committing the configuration update. Otherwise, the Virtual Chassis or VCF could become inconsistent if one or more members have a problem and restart with the new configuration before a planned system reboot activates the change on all members.

Configuring the lpm-profile With Junos OS 13.2x51-D15 and Later

If you use the `lpm-profile` profile with Junos OS 13.2x51-D15 and later, you can control whether the switch allocates any memory for IPv6 prefixes /65 through /127. By default, the switch supports the following with this profile:

- 128K IPv4 prefixes
- 16K IPv6 prefixes (all lengths)

You can disable support for IPv6 prefixes /65 through /127 with the `lpm-profile` profile so that there is more memory for IPv6 prefixes shorter than /65. To do so, enter and commit the following statement:

```
user@switch# set chassis forwarding-options lpm-profile prefix-65-127-disable
```

If you enter this statement, the switch allocates memory for the following:

- 128K IPv4 and IPv6 prefixes shorter than /65

- 0K IPv6 prefixes /65 through /127

For example, if you use the **prefix-65-127-disable** option, each of the following combinations are valid:

- 100K IPv4 and 28K IPv6 /64 prefixes
- 64K IPv4 and 64K IPv6 /64 prefixes
- 128K IPv4 and 0K IPv6 /64 prefixes
- 0K IPv4 and 128K IPv6 /64 prefixes

Configuring the lpm-profile With Junos OS 14.1x53-D30 and Later

If you use the **lpm-profile** profile with Junos OS 14.1x53-D30 or later, you can configure the system to store unicast IPv4 and IPv6 host addresses in the LPM table by using the **unicast-in-lpm** option, thereby freeing memory in the host table. When you use this option, unicast IPv4 and IPv6 addresses are stored in the LPM table instead of the host table, as shown in [Table 31 on page 253](#). You can also use the **prefix-65-127-disable** option to maximize the number of IPv4 addresses and IPv6 addresses with prefixes shorter than /65 (and provide no memory for IPv6 addresses with prefixes longer than /64.)

Table 31: lpm-profile with unicast-in-lpm Option

prefix-65-127-disable?	MAC Table	Host Table (multicast addresses)						LPM Table unicast addresses)		
	MAC	IPv4 unicast	IPv6 unicast	IPv4 (*, G)	IPv4 (S, G)	IPv6 (*, G)	IPv6 (S, G)	IPv4 unicast	IPv6 unicast (</65)	IPv6 unicast (>/64)
No	32K	0	0	8K	8K	4K	4K	128K	16K	16K
Yes	32K	0	0	8K	8K	4K	4K	128K	128K	0

Note that all entries in each table share the same memory space. If a table stores the maximum number of entries for any given type, the entire shared table is full and is unable to accommodate any entries of any other type. For example, if you use the the **unicast-in-lpm** option and there are 128K IPv4 unicast addresses stored in the LPM table, the entire LPM table is full and no IPv6 addresses can be stored. Similarly, if you use the **unicast-in-lpm** option but do not use the **prefix-65-127-disable** option and 16K IPv6 addresses with prefixes shorter than /65 are saved, the entire LPM table is full and no additional addresses (IPv4 or IPv6) can be stored.

To use the **unicast-in-lpm** option, enter and commit the following statement:

```
user@switch# set chassis forwarding-options lpm-profile unicast-in-lpm
```

To use the **prefix-65-127-disable** option, enter and commit the following statement:

```
user@switch# set chassis forwarding-options lpm-profile prefix-65-127-disable
```

Related Documentation

- [Understanding the Unified Forwarding Table on page 245](#)

PART 10

Configuration Statements and Operational Commands

- [Ethernet Ring Protection Configuration Statements on page 257](#)
- [VLAN Configuration Statements on page 269](#)
- [MAC Address Configuration Statements on page 315](#)
- [Private VLAN Configuration Statements on page 321](#)
- [STP Configuration Statements on page 327](#)
- [Q-in-Q Configuration Statements on page 361](#)
- [Reflective Relay Configuration Statement on page 371](#)
- [Unified Forwarding Table Configuration Statements on page 373](#)
- [Bridging and VLANs Monitoring Commands on page 377](#)
- [MAC Address Operational Commands on page 405](#)
- [Spanning Tree Monitoring Commands on page 417](#)

CHAPTER 10

Ethernet Ring Protection Configuration Statements

- [control-channel on page 258](#)
- [control-vlan on page 259](#)
- [data-channel on page 260](#)
- [east-interface on page 261](#)
- [ethernet-ring on page 262](#)
- [guard-interval on page 263](#)
- [hold-interval \(Protection Group\) on page 264](#)
- [protection-group on page 265](#)
- [restore-interval on page 266](#)
- [ring-protection-link-end on page 267](#)
- [ring-protection-link-owner on page 267](#)
- [west-interface on page 268](#)

control-channel

Syntax	<code>control-channel <i>channel-name</i> { vlan <i>vlan-id</i>; interface name <i>interface-name</i> }</code>
Hierarchy Level	[edit protocols protection-group ethernet-ring <i>name</i> (east-interface west-interface)]
Release Information	Statement introduced in Junos OS Release 9.4. Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.
Description	Configure the Ethernet RPS control channel logical interface to carry the RAPS PDU. The related physical interface is the physical ring port.
Options	vlan <i>vlan-id</i> —If the control channel logical interface is a trunk port, then a dedicated vlan <i>vlan-id</i> defines the dedicated VLAN channel to carry the RAPS traffic. Only configure the vlan-id when the control channel logical interface is the trunk port. interface name <i>interface-name</i> —Interface name of the control channel.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Ethernet Ring Protection Switching Overview on page 3• Example: Configuring Ethernet Ring Protection Switching on EX Series Switches• Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12• Configuring Ethernet Ring Protection Switching (CLI Procedure) on page 9

control-vlan

Syntax	control-vlan (<i>vlan-id</i> <i>vlan-name</i>)
Hierarchy Level	[edit protocols protection-group ethernet-ring] [edit protocols protection-group ethernet-ring name (east-interface west-interface)]
Release Information	Statement introduced in Junos OS Release 9.4. Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.
Description	Specify the VLAN that carries the protocol data units (PDUs) between the nodes in the protected Ethernet ring. This is a control VLAN, meaning that it carries data for one instance of an Ethernet ring protection switching (ERPS) in the control channel. Use a control VLAN on trunk port interfaces. One control channel can contain multiple control VLANs.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Example: Configuring Ethernet Ring Protection Switching on EX Series Switches • Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12 • Configuring Ethernet Ring Protection Switching (CLI Procedure) on page 9

data-channel

Syntax	<code>data-channel { vlan <i>number</i>; }</code>
Hierarchy Level	[edit protocols protection-group ethernet-ring <i>ring-name</i>]
Release Information	Statement introduced in Junos OS Release 10.2. Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.
Description	<p>For Ethernet ring protection, configure a data channel to define a set of VLAN IDs that belong to a ring instance.</p> <p>VLANs specified in the data channel use the same topology used by the ERPS PDU in the control channel. Therefore, if a ring interface is blocked in the control channel, all traffic in the data channel is also blocked on that interface.</p>
Options	vlan <i>number</i> —Specify (by VLAN ID) one or more VLANs that belong to a ring instance.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Ethernet Ring Protection Using Ring Instances for Load Balancing</i>• <i>Example: Configuring Load Balancing Within Ethernet Ring Protection for MX Series Routers</i>• Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12• Configuring Ethernet Ring Protection Switching (CLI Procedure) on page 9

east-interface

Syntax

```
east-interface {
  node-id mac-address;
  control-channel channel-name {
    vlan number;
    interface name interface-name
  }
  interface-none
  ring-protection-link-end;
}
```

Hierarchy Level [edit protocols protection-group **ethernet-ring** *ring-name*]

Release Information Statement introduced in Junos OS Release 9.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.

Description Define one of the two interface ports for Ethernet ring protection, the other being defined by the **west-interface** statement at the same hierarchy level. The interface must use the control channel's logical interface name. The control channel is a dedicated VLAN channel for the ring port.

EX Series switches do not use the node-id statement--the node ID is automatically configured on the switches using the MAC address.



NOTE: Always configure this port first, before configuring the **west-interface** statement.



NOTE: The Node ID is not configurable on EX Series switches. The node ID is automatically configured using the MAC address.

The statements are explained separately.

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

Related Documentation

- [Ethernet Ring Protection Switching Overview on page 3](#)
- [Ethernet Ring Protection Using Ring Instances for Load Balancing](#)
- [west-interface on page 268](#)
- [ethernet-ring on page 262](#)
- [Example: Configuring Ethernet Ring Protection Switching on EX Series Switches](#)

- [Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12](#)
- [Configuring Ethernet Ring Protection Switching \(CLI Procedure\) on page 9](#)

ethernet-ring

```
Syntax ethernet-ring ring-name {
    control-vlan (vlan-id | vlan-name);
    data-channel {
        vlan number
    }
    east-interface {
        control-channel channel-name {
            vlan number;
            interface name interface-name
        }
    }
    guard-interval number;
    node-id mac-address;
    restore-interval number;
    ring-protection-link-owner;
    west-interface {
        control-channel channel-name {
            vlan number;
        }
    }
}
```

Hierarchy Level [edit protocols protection-group]

Release Information Statement introduced in Junos OS Release 9.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.

Description For Ethernet PICs on MX Series routers or for EX Series switches, , specify the Ethernet ring in an Ethernet ring protection switching configuration.

Options *ring-name*—Name of the Ethernet protection ring.

The remaining statements are explained separately.

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

Related Documentation

- [Ethernet Ring Protection Switching Overview on page 3](#)
- [Example: Configuring Ethernet Ring Protection Switching on EX Series Switches](#)
- [Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12](#)
- [Configuring Ethernet Ring Protection Switching \(CLI Procedure\) on page 9](#)

guard-interval

Syntax	<code>guard-interval <i>number</i>;</code>
Hierarchy Level	[edit protocols protection-group ethernet-ring <i>ring-name</i>]
Release Information	Statement introduced in Junos OS Release 9.4. Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.
Description	When a link goes down, the ring protection link (RPL) activates. When the downed link comes back up, the RPL link receives notification, restores the link, and waits for the restore interval before issuing another block on the same link. This configuration is a global configuration and applies to all Ethernet rings if the Ethernet ring does not have a more specific configuration for this value. If no parameter is configured at the protection group level, the global configuration of this parameter uses the default value.
Options	<i>number</i> —Guard timer interval, in milliseconds. Range: 10 through 2000 ms Default: 500 ms
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Ethernet Ring Protection Switching Overview on page 3 • Example: Configuring Ethernet Ring Protection Switching on EX Series Switches • Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12 • Configuring Ethernet Ring Protection Switching (CLI Procedure) on page 9

hold-interval (Protection Group)

Syntax	hold-interval <i>number</i> ;
Hierarchy Level	[edit protocols protection-group ethernet-ring name]
Release Information	Statement introduced in Junos OS Release 9.4.
Description	Specify the hold-off timer interval <i>for all rings</i> in 100 millisecond (ms) increments.
Options	<p><i>number</i>—Hold-timer interval, in milliseconds.</p> <p>Range: 0 through 10,000 ms</p> <p>Default: 100 ms</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Ethernet Ring Protection Switching Overview on page 3• Example: Configuring Ethernet Ring Protection Switching on EX Series Switches• Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12

protection-group

```
Syntax  protection-group {
        ethernet-ring ring-name {
            control-vlan (vlan-id | vlan-name);
            data-channel {
                vlan number
            }
            east-interface {
                control-channel channel-name {
                    vlan number;
                    interface name interface-name
                }
            }
            guard-interval number;
            hold-interval number;
            node-id mac-address;
            restore-interval number;
            ring-protection-link-owner RPL owner flag;
            west-interface {
                control-channel channel-name {
                    vlan number;
                    interface name interface-name
                }
            }
            guard-interval number;
            hold-interval
            node-id mac-address;
            restore-interval number;
            traceoptions {
                file filename <no-stamp> <world-readable | no-world-readable> <replace> <size size>;
                flag flag;
            }
        }
    }
```

Hierarchy Level [edit protocols]

Release Information Statement introduced in Junos OS Release 12.1 for EX Series switches.

Description Configure Ethernet ring protection switching (ERPS).

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

Related Documentation

- [Example: Configuring Ethernet Ring Protection Switching on EX Series Switches](#)
- [Configuring Ethernet Ring Protection Switching \(CLI Procedure\) on page 9](#)
- [Ethernet Ring Protection Switching Overview on page 3](#)

restore-interval

Syntax	<code>restore-interval <i>number</i>;</code>
Hierarchy Level	[edit protocols protection-group ethernet-ring <i>ring-name</i>]
Release Information	Statement introduced in Junos OS Release 9.4. Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.
Description	Configures the number of minutes that the node does not process any Ethernet ring protection (ERP) protocol data units (PDUs).. This configuration is a global configuration and applies to all Ethernet rings if the Ethernet ring does not have a more specific configuration for this value. If no parameter is configured at the protection group level, the global configuration of this parameter uses the default value.
Options	<i>number</i> —Specify the restore interval. Range: 5 through 12 minutes
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Ethernet Ring Protection Switching Overview on page 3• Example: Configuring Ethernet Ring Protection Switching on EX Series Switches• Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12• Configuring Ethernet Ring Protection Switching (CLI Procedure) on page 9

ring-protection-link-end

Syntax	ring-protection-link-end;
Hierarchy Level	[edit protocols protection-group ethernet-ring <i>ring-name</i> (east-interface west-interface)]
Release Information	Statement introduced in Junos OS Release 9.4. Statement introduced in Junos OS Release 12.1 for EX Series switches.
Description	Specify that the port is one side of a ring protection link (RPL) by setting the RPL end flag.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Ethernet Ring Protection Switching Overview on page 3 • Example: Configuring Ethernet Ring Protection Switching on EX Series Switches • Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12 • Configuring Ethernet Ring Protection Switching (CLI Procedure) on page 9

ring-protection-link-owner

Syntax	ring-protection-link-owner;
Hierarchy Level	[edit protocols protection-group ethernet-ring <i>ring-name</i>]
Release Information	Statement introduced in Junos OS Release 9.4. Statement introduced in Junos OS Release 12.1 for EX Series switches. Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.
Description	Specify the ring protection link (RPL) owner flag in the Ethernet protection ring. Include this statement only once for each ring (only one node can function as the RPL owner).
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Ethernet Ring Protection Switching Overview on page 3 • Example: Configuring Ethernet Ring Protection Switching on EX Series Switches • Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12

west-interface

Syntax

```
west-interface {
  node-id mac-address;
  control-channel channel-name {
    vlan number;
    interface name interface-name
  }
  interface-name
  ring-protection-link-end;
}
```

Hierarchy Level [edit protocols protection-group **ethernet-ring** ring-name]

Release Information Statement introduced in Junos OS Release 9.5.
Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 14.153-D10 for QFX Series switches.

Description Define one of the two interface ports for Ethernet ring protection, the other being defined by the **east-interface** statement at the same hierarchy level. The interface must use the control channel's logical interface name. The control channel is a dedicated VLAN channel for the ring port.



NOTE: Always configure this port second, after configuring the **east-interface** statement.

The statements are explained separately.

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

Related Documentation

- [Ethernet Ring Protection Switching Overview on page 3](#)
- [Ethernet Ring Protection Using Ring Instances for Load Balancing](#)
- [east-interface on page 261](#)
- [ethernet-ring on page 262](#)
- [Example: Configuring Ethernet Ring Protection Switching on EX Series Switches](#)
- [Example: Configuring Ethernet Ring Protection Switching on QFX Series and EX Series Switches Supporting ELS on page 12](#)
- [Configuring Ethernet Ring Protection Switching \(CLI Procedure\) on page 9](#)

CHAPTER 11

VLAN Configuration Statements

- [\[edit vlans\] Configuration Statement Hierarchy on the QFX Series on page 270](#)
- [autostate-exclude on page 273](#)
- [description \(VLAN\) on page 274](#)
- [dhcp-relay on page 275](#)
- [filter \(VLANs\) on page 280](#)
- [forwarding-options on page 281](#)
- [interface \(VLANs\) on page 286](#)
- [interface-mac-limit on page 287](#)
- [interface-mode on page 289](#)
- [irb \(Interfaces\) on page 291](#)
- [isolated-vlan on page 294](#)
- [l3-interface \(VLAN\) on page 295](#)
- [mac \(Static MAC-Based VLANs\) on page 296](#)
- [members on page 297](#)
- [native-vlan-id on page 298](#)
- [packet-action on page 299](#)
- [port-mode on page 302](#)
- [private-vlan on page 303](#)
- [service-id on page 304](#)
- [switch-options on page 305](#)
- [static \(Static MAC-Based VLANs\) on page 306](#)
- [static-mac on page 306](#)
- [vlan-id \(VLANs\) on page 307](#)
- [vlan-id-list on page 308](#)
- [vlan-rewrite on page 309](#)
- [vlan-tagging on page 310](#)
- [vlan-tags on page 311](#)
- [vlans on page 312](#)

[edit vlans] Configuration Statement Hierarchy on the QFX Series

This topic lists supported and unsupported configuration statements in the **[edit vlans]** hierarchy level on EX Series switches.

- *Supported* statements are those that you can use to configure some aspect of a software feature on the switch.
- *Unsupported* statements are those that appear in the command-line interface (CLI) on the switch, but that have no effect on switch operation if you configure them.
- Not all features are supported on all switch platforms. For detailed information about feature support on specific EX Series switch platforms, see *QFX Series Virtual Chassis Software Features Overview*.

This topic lists:

- [Supported Statements in the \[edit vlans\] Hierarchy Level on page 270](#)
- [Unsupported Statements in the \[edit vlans\] Hierarchy Level on page 272](#)

Supported Statements in the [edit vlans] Hierarchy Level

The following hierarchy shows the **[edit vlans]** configuration statements supported on one or more of the EX Series switches:

```
vlan {
  vlan-name {
    description text-description;
    domain-type bridge;
    forwarding-options {
      dhcp-security {
        arp-inspection;
        group group-name {
          interface interface-name {
            static-ip ip-address {
              mac mac-address;
            }
          }
        }
        overrides {
          no-option82;
          trusted;
        }
      }
    }
    ip-source-guard;
    no-dhcp-snooping;
    option-82 {
      circuit-id {
        prefix {
          host-name;
          logical-system-name;
          routing-instance-name;
        }
      }
      use-interface-description (device | logical);
      use-vlan-id;
    }
  }
}
```

```

    }
    remote-id {
        host-name;
        use-interface-description (device | logical);
        use-string string;
    }
    vendor-id {
        use-string string;
    }
}
}
filter {
    input filter-name;
    output filter-name;
}
flood {
    input filter-name;
}
}
}
l3-interface irb.logical-unit-number;
multicast-snooping-options {
    flood-groups [group-names];
    forwarding-cache {
        threshold {
            reuse threshold;
            suppress threshold;
        }
    }
    graceful-restart {
        disable;
        restart-duration duration;
    }
    host-outbound-traffic {
        dot1p bits;
        forwarding-class forwarding-class;
    }
    multichassis-lag-replicate-state;
    nexthop-hold-time time;
    options {
        syslog {
            level level;
            mark interval;
            upto level;
        }
    }
}
traceoptions {
    file filename {
        files number;
        no-world-readable;
        size file-size;
        world-readable;
    }
    flag flag {
        disable;
    }
}
}

```

```

}
switch-options {
  interface interface-name {
    interface-mac-limit limit {
      packet-action action;
    }
    static-mac mac-address;
  }
  interface-mac-limit limit {
    packet-action action;
  }
  mac-move-limit limit {
    packet-action action;
  }
  mac-table-size limit {
    packet-action drop;
  }
  no-mac-learning;
}
vlan-id number;
vlan-id-list [vlan-id | vlan-id-vlan-id];
}
}

```

Unsupported Statements in the [edit vlans] Hierarchy Level

All statements in the [edit vlans] hierarchy level that are displayed in the command-line interface (CLI) on the switch are supported on the switch and operate as documented with the following exceptions:

Table 32: Unsupported [edit vlans] Configuration Statements on EX Series Switches

Statement	Hierarchy Level
NOTE: Variables, such as <i>filename</i> , are not shown in the statements or hierarchies.	
mcae-mac-synchronize	[edit vlans]
no-irb-layer-2-copy	[edit vlans]

Related Documentation

- [Understanding Bridging and VLANs on page 35](#)

autostate-exclude

Syntax	autostate-exclude;
Hierarchy Level	[edit interface <i>interface-name</i> ether-options]
Release Information	Statement introduced in Junos OS Release 14.1x53-D40 on QFX5100 switches.
Description	<p>Specify not to include a routed VLAN interface (RVI) in the state calculation for VLAN members. The default behavior is not to exclude an RVI in the state calculation unless all the ports on the interface go down. Because an RVI often has multiple ports in a single VLAN, the state calculation for a VLAN member might include a port that is down, possibly resulting in traffic loss. This feature enables you to exclude a trunk or access interface from the state calculation, which results in the RVI interface being marked as down as soon as the port specifically assigned to a VLAN goes down.</p> <p>RVIs are used to bind specific VLANs to Layer 3 interfaces, enabling a switch to forward packets between those VLANs— without having to configure another device, such as a router, to connect VLANs. In a typical scenario, a port on the interface is assigned to a specific VLAN, while a different port on that interface is assigned to an 802.1Q trunk interface to carry traffic between multiple VLANs, and a third port on that interface is assigned to an access interface used to connect the VLAN to network devices.</p> <p>To ensure that an interface is marked as down and thereby excluded from the state calculation for VLAN members when the port assigned to the VLAN goes down, configure this statement on the trunk or access interface. The trunk or access interface is automatically excluded from the state calculation of the RVI. In this way, when a port assigned to a specified VLAN goes down, the RVI assigned to that VLAN is also marked as down.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration
Related Documentation	<ul style="list-style-type: none"> • Excluding a Routed VLAN Interface from State Calculations on page 79 • port-mode on page 302 • show ethernet-switching interface on page 380

description (VLAN)

Syntax	<code>description <i>text-description</i>;</code>
Hierarchy Level	[edit vlans <i>vlan-name</i>]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	Provide a textual description for the VLAN. The text has no effect on the operation of the VLAN or switch.
Options	<i>text-description</i> —Text to describe the interface. It can contain letters, numbers, and hyphens (-) and can be up to 255 characters long. If the text includes spaces, enclose the entire text 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	<ul style="list-style-type: none">• Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43• Understanding Bridging and VLANs on page 35• show vlans on page 396

dhcp-relay

```

Syntax  dhcp-relay {
        active-server-group server-group-name;
        authentication {
            password password-string;
            username-include {
                circuit-type;
                delimiter delimiter-character;
                domain-name domain-name-string;
                interface-name;
                logical-system-name;
                mac-address;
                option-60;
                option-82 <circuit-id> <remote-id>;
                routing-instance-name;
                user-prefix user-prefix-string;
            }
        }
        dhcpv6 {
            active-server-group server-group-name;
            authentication {
                password password-string;
                username-include {
                    circuit-type;
                    client-id;
                    delimiter delimiter-character;
                    domain-name domain-name-string;
                    interface-name;
                    logical-system-name;
                    relay-agent-interface-id;
                    relay-agent-remote-id;
                    relay-agent-subscriber-id;
                    routing-instance-name;
                    user-prefix user-prefix-string;
                }
            }
            forward-only-replies;
            dynamic-profile profile-name {
                aggregate-clients (merge | replace);
                use-primary primary-profile-name;
            }
        }
        forward-only {
            routing-instance (current | default | routing-instance-name);
        }
        forward-only-replies;
        group group-name {
            active-server-group server-group-name;
            authentication {
                ...
            }
            dynamic-profile profile-name {
                ...
            }
        }
    }

```

```

}
forward-only {
    routing-instance (current | default | routing-instance-name);
}
interface interface-name {
    exclude;
    liveness-detection {
        failure-action (clear-binding | clear-binding-if-interface-up | log-only);
        method {
            bfd {
                version (0 | 1 | automatic);
                minimum-interval milliseconds;
                minimum-receive-interval milliseconds;
                multiplier number;
                no-adaptation;
                transmit-interval {
                    minimum-interval milliseconds;
                    threshold milliseconds;
                }
                detection-time {
                    threshold milliseconds;
                }
                session-mode (automatic | multihop | singlehop);
                holddown-interval milliseconds;
            }
        }
    }
}
overrides {
    ...
}
service-profile dynamic-profile-name;
trace;
upto upto-interface-name;
}
service-profile dynamic-profile-name;
}
overrides {
    ...
}
relay-agent-interface-id {
    ...
}
service-profile dynamic-profile-name;
liveness-detection {
    failure-action (clear-binding | clear-binding-if-interface-up | log-only);
    method {
        bfd {
            version (0 | 1 | automatic);
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            multiplier number;
            no-adaptation;
            transmit-interval {
                minimum-interval milliseconds;
                threshold milliseconds;
            }
        }
    }
}

```

```

        detection-time {
            threshold milliseconds;
        }
        session-mode(automatic | multihop | singlehop);
        holddown-interval milliseconds;
    }
}
}
overrides {
    allow-snooped-clients;
    interface-client-limit number;
    no-allow-snooped-clients;
    no-bind-on-request;
    send-release-on-delete;
}
relay-agent-interface-id {
    prefix prefix;
    use-interface-description (logical | device);
}
server-group {
    server-group-name {
        server-ip-address;
    }
}
dynamic-profile profile-name {
    aggregate-clients (merge | replace);
    use-primary primary-profile-name;
}
forward-snooped-clients (all-interfaces | configured-interfaces |
    non-configured-interfaces);
group group-name {
    active-server-group server-group-name;
    authentication {
        ...
    }
}
dynamic-profile profile-name {
    ...
}
interface interface-name {
    exclude;
    liveness-detection {
        failure-action (clear-binding | clear-binding-if-interface-up | log-only);
        method {
            bfd {
                version (0 | 1 | automatic);
                minimum-interval milliseconds;
                minimum-receive-interval milliseconds;
                multiplier number;
                no-adaptation;
                transmit-interval {
                    minimum-interval milliseconds;
                    threshold milliseconds;
                }
            }
            detection-time {
                threshold milliseconds;
            }
        }
    }
}

```

```

        session-mode(automatic | multihop | singlehop);
        holddown-interval milliseconds;
    }
}
overrides {
    ...
}
service-profile dynamic-profile-name;
trace;
upto upto-interface-name;
}
overrides {
    ...
}
relay-option-82 {
    ...
}
service-profile dynamic-profile-name;
}
liveness-detection {
    failure-action (clear-binding | clear-binding-if-interface-up | log-only);
    method {
        bfd {
            version (0 | 1 | automatic);
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            multiplier number;
            no-adaptation;
            transmit-interval {
                minimum-interval milliseconds;
                threshold milliseconds;
            }
            detection-time {
                threshold milliseconds;
            }
            session-mode(automatic | multihop | singlehop);
            holddown-interval milliseconds;
        }
    }
}
overrides {
    allow-snooped-clients;
    always-write-giaddr;
    always-write-option-82;
    client-discover-match <option60-and-option82>;
    disable-relay;
    interface-client-limit number;
    layer2-unicast-replies;
    no-allow-snooped-clients;
    no-bind-on-request;
    proxy-mode;
    replace-ip-source-with;
    send-release-on-delete;
    trust-option-82;
}

```

```

relay-option-82 {
  circuit-id {
    prefix prefix;
    use-interface-description (logical | device);
  }
}
server-group {
  server-group-name {
    server-ip-address;
  }
}
service-profile dynamic-profile-name;
}

```

Hierarchy Level [edit forwarding-options],
[edit vlans forwarding-options]

Release Information Statement introduced in Junos OS Release 11.3 for the QFX Series.

Description Configure extended Dynamic Host Configuration Protocol (DHCP) relay and DHCPv6 relay options on the switch and enable the switch to function as a DHCP relay agent. A DHCP relay agent forwards DHCP request and reply packets between a DHCP client and a DHCP server.

DHCP relay supports the attachment of dynamic profiles and also interacts with the local AAA Service Framework to use back-end authentication servers, such as RADIUS, to provide subscriber authentication. You can attach dynamic profiles and configure authentication support on a global basis or for a specific group of interfaces.

The extended DHCP and DHCPv6 relay agent options configured with the **dhcp-relay** and **dhcpv6** statements are incompatible with the DHCP/BOOTP relay agent options configured with the **bootp** statement. As a result, the extended DHCP or DHCPv6 relay agent and the DHCP/BOOTP relay agent cannot both be enabled on the router at the same time.

The remaining statements are explained separately.

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

Related Documentation

- *Configuring DHCP and BOOTP*
- *Example: Configuring DHCP Message Exchange Between DHCP Server and Clients in Different Virtual Routing Instances to Increase Security of the DHCP Server*

filter (VLANs)

Syntax	<code>filter (input output) <i>filter-name</i>;</code>
Hierarchy Level	<code>[edit vlan <i>vlan-name</i>]</code> <code>[edit vlan <i>vlan-name</i> forwarding-options]</code>
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	Apply a firewall filter to traffic ingressing or egressing a VLAN.
Default	All incoming traffic is accepted unmodified to a VLAN, and all outgoing traffic is sent unmodified from a VLAN.
Options	<i>filter-name</i> —Name of a firewall filter defined at the <code>[edit firewall family <i>family-name</i> filter]</code> hierarchy level. input —Apply a firewall filter to VLAN ingress traffic. output —Apply a firewall filter to VLAN egress traffic.
Required Privilege Level	interface —To view this statement in the configuration. interface-control —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring Firewall Filters</i>• <i>Overview of Firewall Filters</i>

forwarding-options

```
Syntax forwarding-options {
  dhcp-relay {
    active-server-group server-group-name;
    authentication {
      password password-string;
      username-include {
        circuit-type;
        delimiter delimiter-character;
        domain-name domain-name-string;
        interface-name;
        logical-system-name;
        mac-address;
        option-60;
        option-82 <circuit-id> <remote-id>;
        routing-instance-name;
        user-prefix user-prefix-string;
      }
    }
  }
  dhcpv6 {
    active-server-group server-group-name;
    authentication {
      password password-string;
      username-include {
        circuit-type;
        client-id;
        delimiter delimiter-character;
        domain-name domain-name-string;
        interface-name;
        logical-system-name;
        relay-agent-interface-id;
        relay-agent-remote-id;
        relay-agent-subscriber-id;
        routing-instance-name;
        user-prefix user-prefix-string;
      }
    }
  }
  dynamic-profile profile-name {
    aggregate-clients (merge | replace);
    use-primary primary-profile-name;
  }
  forward-only {
    routing-instance (current | default | routing-instance-name);
  }
  forward-only-replies;
  group group-name {
    active-server-group server-group-name;
    authentication {
      ...
    }
    dynamic-profile profile-name {
      ...
    }
  }
}
```

```

interface interface-name {
  exclude;
  liveness-detection {
    failure-action (clear-binding | clear-binding-if-interface-up | log-only);
    method {
      bfd {
        version (0 | 1 | automatic);
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        multiplier number;
        no-adaptation;
        transmit-interval {
          minimum-interval milliseconds;
          threshold milliseconds;
        }
        detection-time {
          threshold milliseconds;
        }
        session-mode (automatic | multihop | singlehop);
        holddown-interval milliseconds;
      }
    }
  }
  overrides {
    ...
  }
  service-profile dynamic-profile-name;
  trace;
  upto upto-interface-name;
}
service-profile dynamic-profile-name;
}
overrides {
  ...
}
relay-agent-interface-id {
  ...
}
service-profile dynamic-profile-name;
liveness-detection {
  failure-action (clear-binding | clear-binding-if-interface-up | log-only);
  method {
    bfd {
      version (0 | 1 | automatic);
      minimum-interval milliseconds;
      minimum-receive-interval milliseconds;
      multiplier number;
      no-adaptation;
      transmit-interval {
        minimum-interval milliseconds;
        threshold milliseconds;
      }
      detection-time {
        threshold milliseconds;
      }
      session-mode (automatic | multihop | singlehop);
    }
  }
}

```



```

        holddown-interval milliseconds;
    }
}
overrides {
    allow-snooped-clients;
    interface-client-limit number;
    no-allow-snooped-clients;
    no-bind-on-request;
    send-release-on-delete;
}
relay-agent-interface-id {
    prefix prefix;
    use-interface-description (logical | device);
}
server-group {
    server-group-name {
        server-ip-address;
    }
}
...;
dynamic-profile profile-name {
    aggregate-clients (merge | replace);
    use-primary primary-profile-name;
}
forward-snooped-clients (all-interfaces | configured-interfaces |
    non-configured-interfaces);
group group-name {
    active-server-group server-group-name;
    authentication {
        ...
    }
}
dynamic-profile profile-name {
    ...
}
interface interface-name {
    exclude;
    liveness-detection {
        failure-action (clear-binding | clear-binding-if-interface-up | log-only);
        method {
            bfd {
                version (0 | 1 | automatic);
                minimum-interval milliseconds;
                minimum-receive-interval milliseconds;
                multiplier number;
                no-adaptation;
                transmit-interval {
                    minimum-interval milliseconds;
                    threshold milliseconds;
                }
            }
            detection-time {
                threshold milliseconds;
            }
        }
        session-mode (automatic | multihop | singlehop);
        holddown-interval milliseconds;
    }
}

```

```

    }
  }
  overrides {
    ...
  }
  service-profile dynamic-profile-name;
  trace;
  upto upto-interface-name;
}
overrides {
  ...
}
... {
  ...
}
relay-option-82 {
  ...
}
service-profile dynamic-profile-name;
}
liveness-detection {
  failure-action (clear-binding | clear-binding-if-interface-up | log-only);
  method {
    bfd {
      version (0 | 1 | automatic);
      minimum-interval milliseconds;
      minimum-receive-interval milliseconds;
      multiplier number;
      no-adaptation;
      transmit-interval {
        minimum-interval milliseconds;
        threshold milliseconds;
      }
      detection-time {
        threshold milliseconds;
      }
      session-mode (automatic | multihop | singlehop);
      holddown-interval milliseconds;
    }
  }
}
overrides {
  allow-snooped-clients;
  always-write-giaddr;
  always-write-option-82;
  client-discover-match <option60-and-option82>;
  disable-relay;
  interface-client-limit number;
  layer2-unicast-replies;
  no-allow-snooped-clients;
  no-bind-on-request;
  proxy-mode;
  replace-ip-source-with;
  send-release-on-delete;
  trust-option-82;
}

```

```

relay-option-82 {
  circuit-id {
    prefix prefix;
    use-interface-description (logical | device);
  }
}
server-group {
  server-group-name {
    server-ip-address;
  }
}
service-profile dynamic-profile-name;
}
dhcp-security {
  arp-inspection;
  group group-name {
    interface interface-name {
      static-ip ip-address {
        mac mac-address;
      }
    }
    overrides {
      no-option82;
      trusted;
      untrusted;
    }
  }
}
ip-source-guard;
no-dhcp-snooping;
option-82 {
  circuit-id {
    prefix {
      host-name;
      logical-system-name;
      routing-instance-name;
    }
    use-interface-description (device | logical);
    use-vlan-id;
  }
  remote-id {
    host-name hostname;
    use-interface-description (device | logical);
    use-string string;
  }
  vendor-id {
    use-string string;
  }
}
}
fip-security {
  examine-vn2vf;
  examine-vn2vn {
    beacon-period milliseconds;
  }
  fc-map fc-map-value;
  interface interface-name {

```

```
        (fcoe-trusted | no-fcoe-trusted;)
    }
}
```

Hierarchy Level [edit]
[edit vlans]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 11.3 for QFX Series switches.

Description Configure traffic forwarding.

The statements are explained separately.

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

interface (VLANs)

Syntax interface *interface-name* {
 mapping (native (push | swap) | tag (push | swap));
}

Hierarchy Level [edit [vlans](#) *vlan-name*]

Release Information Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description For a specific VLAN, configure an interface.

Options *interface-name*—Name of the interface.

The remaining statement is explained separately.

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

Related Documentation

- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Configuring VLANs](#)
- [Understanding Bridging and VLANs](#)

interface-mac-limit

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



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



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



NOTE: The `interface-mac-limit` statement is not supported on the QFX Series.

Default	For an access port, the default MAC limit is 1024 MAC addresses. For a trunk port, the default MAC limit is 8192 MAC addresses.
Options	<p>limit—Maximum number of MAC addresses learned from an interface.</p> <p>Range: 1 through 131,071 MAC addresses per interface</p> <p>The remaining statement is explained separately.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Layer 2 Learning and Forwarding for Bridge Domains Overview• Layer 2 Learning and Forwarding for VLANs Overview on page 34• Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports• Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port

interface-mode

Syntax	interface-mode (access trunk);
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family bridge], [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family ethernet-switching], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family bridge]
Release Information	Statement introduced in Junos OS Release 9.2. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Statement introduced in Junos OS Release 13.2 for the QFX Series.
Description	<p> NOTE: This statement supports the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see port-mode. For ELS details, see <i>Getting Started with Enhanced Layer 2 Software</i>.</p> <p>(QFX Series 3500 and 3600 standalone switches)—Determine whether the logical interface accepts or discards packets based on VLAN tags. Specify the trunk option to accept packets with a VLAN ID that matches the list of VLAN IDs specified in the vlan-id or vlan-id-list statement, then forward the packet within the bridge domain or VLAN configured with the matching VLAN ID. Specify the access option to accept packets with no VLAN ID, then forward the packet within the bridge domain or VLAN configured with the VLAN ID that matches the VLAN ID specified in the vlan-id statement.</p> <p> NOTE: On MX Series routers, if you want IGMP snooping to be functional for a bridge domain, then you should not configure interface-mode and irb for that bridge. Such a configuration commit succeeds, but IGMP snooping is not functional, and a message informing the same is displayed. For more information, see <i>Configuring a Trunk Interface on a Bridge Network</i>.</p>
Options	<p>access—Configure a logical interface to accept untagged packets. Specify the VLAN to which this interface belongs using the vlan-id statement.</p> <p>trunk—Configure a single logical interface to accept packets tagged with any VLAN ID specified with the vlan-id or vlan-id-list statement.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Configuring a Logical Interface for Access Mode</i> • <i>Configuring a Logical Interface for Trunk Mode</i>

- *Example: Connecting Access Switches to a Distribution Switch*

irb (Interfaces)

```

Syntax  irb {
    accounting-profile name;
    arp-l2-validate;
    description text;

    (gratuitous-arp-reply | no-gratuitous-arp-reply);
    hold-time up milliseconds down milliseconds;
    mtu bytes;
    no-gratuitous-arp-request;

    traceoptions {
        flag flag;
    }
    (traps | no-traps);
    unit logical-unit-number {
        accounting-profile name;
        bandwidth rate;
        description text;
        disable;
        encapsulation type;
        family inet {
            accounting {
                destination-class-usage;
                source-class-usage {
                    input;
                    output;
                }
            }
        }
        address ipv4-address {
            arp ip-address (mac | multicast-mac) mac-address <publish>;
            broadcast address;
            preferred;
            primary;
            vrrp-group group-number {
                (accept-data | no-accept-data);
                advertise-interval seconds;
                advertisements-threshold number;
                authentication-key key;
                authentication-type authentication;
                fast-interval milliseconds;
                (preempt | no-preempt) {
                    hold-time seconds;
                }
                priority number;
                track {
                    interface interface-name {
                        bandwidth-threshold bandwidth;
                        priority-cost number;
                    }
                    priority-hold-time seconds;
                    route ip-address/mask routing-instance instance-name priority-cost cost;
                }
            }
        }
    }
}

```

```
    virtual-address [ addresses ];
    vrrp-inherit-from {
        active-group group-number;
        active-interface interface-name;
    }
}
filter {
    input filter-name;
    output filter-name;
}
mtu bytes;
no-neighbor-learn;
no-redirects;
primary;
rpf-check {
    fail-filter filter-name;
    mode {
        loose;
    }
}
targeted-broadcast {
    forward-and-send-to-re;
    forward-only;
}
}
family inet6 {
    accounting {
        destination-class-usage;
        source-class-usage {
            input;
            output;
        }
    }
}
address address {
    eui-64;
    ndp ip-address (mac | multicast-mac) mac-address <publish>;
    preferred;
    primary;
    vrrp-inet6-group group-id {
        accept-data | no-accept-data;
        advertisements-threshold number;
        authentication-key key;
        authentication-type authentication;
        fast-interval milliseconds;
        inet6-advertise-interval milliseconds;
        preempt | no-preempt {
            hold-time seconds;
        }
        priority number;
        track {
            interface interface-name {
                bandwidth-threshold bandwidth priority-cost number;
                priority-cost number;
            }
            priority-hold-time seconds;
        }
    }
}
```

```


        route ip-address/mask routing-instance instance-name priority-cost cost;
    }
    virtual-inet6-address [addresses];
    virtual-link-local-address ipv6-address;
    vrrp-inherit-from {
        active-group group-number;
        active-interface interface-name;
    }
}
}
(dad-disable | no-dad-disable);
filter {
    input filter-name;
    output filter-name;
}
mtu bytes;
nd6-stale-time seconds;
no-neighbor-learn;
no-redirects;
policer {
    input policer-name;
    output policer-name;
}
rpf-check {
    fail-filter filter-name;
    mode {
        loose;
    }
}
}
family iso {
    address interface-address;
    mtu bytes;
}
family mpls {
    filter {
        input filter-name;
        output filter-name;
    }
    mtu bytes;
    policer {
        input policer-name;
        output policer-name;
    }
}
native-inner-vlan-id vlan-id;
proxy-arp (restricted | unrestricted);
(traps | no-traps);
vlan-id-list [vlan-id's];
vlan-id-range [vlan-id-range];
}
}

```

Hierarchy Level [edit interfaces *interface-name*

Release Information	Statement introduced in Junos OS Release 12.3R2 for EX Series switches. irb option introduced in Junos OS Release 13.2 for the QFX Series.
Description	Configure the properties of a specific integrated bridging and routing (IRB) interface. The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• [edit interfaces] Hierarchy Level• [edit interfaces] Configuration Statement Hierarchy on EX Series Switches

isolated-vlan

Syntax	<code>isolated-vlan vlan-name <i>isolated-vlan-name</i> vlan-id <i>isolated-vlan-id</i>;</code>
Hierarchy Level	<code>[edit vlans <i>primary-vlan-name</i> vlan-id <i>primary-vlan-vlan-id</i>]</code>
Release Information	Statement introduced in Junos OS Release 14.1X53-D10 for EX Series switches. Statement introduced in Junos OS Release 14.1X53-D15 for QFX Series switches.
Description	Configure the specified isolated VLAN to be a secondary VLAN of the specified primary VLAN. An isolated VLAN receives packets only from the primary VLAN and forwards frames upstream to the primary VLAN.
<div> NOTE: Before you specify this configuration statement, you must have already configured an isolated VLAN and assigned a VLAN ID to it. See private-vlan.</div>	
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Creating a Private VLAN on a Single Switch (CLI Procedure) on page 214• Creating a Private VLAN Spanning Multiple Switches (CLI Procedure) on page 216

l3-interface (VLAN)

Syntax	<code>l3-interface (vlan.logical-interface-number irb.logical-interface-number);</code>
Hierarchy Level	[edit vlan <i>vlan-name</i>]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series. irb option introduced in Junos OS Release 13.2 for the QFX Series.
Description	Associate a Layer 3 interface with the VLAN. Configure Layer 3 interfaces on trunk ports to allow the interface to transfer traffic between VLANs. Traffic between VLANs must be routed, which requires a common Layer 3 interface.
Default	No Layer 3 (routing) interface is associated with the VLAN.
Options	<code>vlan.logical-interface-number</code> —Number of the logical interface. Use the unit number that you used when you created the vlan interface with a set interfaces vlan unit statement.



NOTE: Use this statement with versions of Junos OS that do not support Enhanced Layer 2 Software (ELS).

`irb.logical-interface-number`—Logical interface defined with a **set interfaces irb** statement.



NOTE: Use this statement with versions of Junos OS that support Enhanced Layer 2 Software (ELS).

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

- Related Documentation**
- [show ethernet-switching interfaces on page 383](#)
 - [show vlans on page 396](#)

mac (Static MAC-Based VLANs)

Syntax	<code>mac <i>mac-address</i> { next-hop <i>interface-name</i>; }</code>
Hierarchy Level	<code>[edit ethernet-switching-options static vlan <i>vlan-name</i>]</code>
Description	<p>Specify the MAC address to add to the Ethernet switching table.</p> <p>The remaining statement is explained separately.</p>
Options	<i>mac-address</i> —MAC address
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• <i>Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure)</i>

members

Syntax	<code>members [(all <i>names</i> <i>vlan-ids</i>)];</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> unit 0 family ethernet-switching vlan]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	For trunk interfaces, configure the VLANs for which the interface can carry traffic.



TIP: To display a list of all configured VLANs on the system, including VLANs that are configured but not committed, type ? after `vlan` or `vlands` in your configuration mode command line. Note that only one VLAN is displayed for a VLAN range.

Options `all`—Specify that this trunk interface be a member of all the VLANs that are configured on this switch. When a new VLAN is configured on the switch, this trunk interface automatically becomes a member of the VLAN.



NOTE: Each VLAN that is configured must have a specified VLAN ID when you attempt to commit the configuration; otherwise, the configuration commit fails. Also, `all` cannot be the name of a VLAN on the switch.

names—Names of one or more VLANs.

vlan-ids—Numeric identifiers of one or more VLANs.

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

Related Documentation

- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Understanding Bridging and VLANs on page 35](#)
- [show ethernet-switching interfaces on page 383](#)
- [show vlans on page 396](#)

native-vlan-id

Syntax	<code>native-vlan-id <i>vlan-id</i>;</code>
Hierarchy Level	For platforms without ELS: <code>[edit interfaces <i>interface-name</i> unit 0 family ethernet-switching],</code> For platforms with ELS: <code>[edit interfaces <i>interface-name</i>]</code>
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	<p>Configure the VLAN identifier to associate with untagged packets received on the interface. The logical interface on which untagged packets are received must be configured with the same VLAN ID as the native VLAN ID configured on the physical interface. To configure the logical interface, include the vlan-id statement (matching the native-vlan-id statement on the physical interface) at the <code>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]</code> hierarchy level.</p> <p>When the native-vlan-id statement is combined with the interface-mode statement, untagged packets are accepted and forwarded within the bridge domain or VLAN that is configured with the matching VLAN ID.</p> <p>When the native-vlan-id statement is combined with the flexible-vlan-tagging statement, untagged packets are accepted on the interfaces that are configured for Q-in-Q tunneling.</p> <p>.</p>
Options	<p>vlan-id—Numeric identifier of the VLAN.</p> <p>Range: 1 through 4094</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• Junos OS Network Interfaces Configuration Guide•• show ethernet-switching interfaces on page 383• show vlans on page 396

packet-action

Syntax `packet-action action;`

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

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

Support for logical systems added in Junos OS Release 9.6.

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

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

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

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

Default



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

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

Options

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

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

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

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


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




NOTE: On EX4300 switches, the shutdown option causes an interface to stop learning MAC addresses and it also drops all incoming packets, but does not disable the physical interface.

Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring EVPN Routing Instances</i>• Configuring MAC Limiting (CLI Procedure) on page 94• <i>Configuring Persistent MAC Learning (CLI Procedure)</i>• <i>Layer 2 Learning and Forwarding for Bridge Domains Overview</i>• Layer 2 Learning and Forwarding for VLANs Overview on page 34• <i>Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports</i>• Layer 2 Learning and Forwarding for VLANs Overview on page 34• <i>Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port</i>

port-mode

Syntax	<code>port-mode (access tagged-access trunk);</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family ethernet-switching]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	<div> NOTE: This statement does not support the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that supports ELS, see interface-mode. For ELS details, see <i>Getting Started with Enhanced Layer 2 Software</i>.</div> <p>Configure whether an interface on the switch operates in access, tagged access, or trunk mode.</p>
Default	All switch interfaces are in access mode.
Options	<p>access—Have the interface operate in access mode. In this mode, the interface can be in a single VLAN only. Access interfaces typically connect to network devices, such as PCs, printers, IP telephones, and IP cameras.</p> <p>tagged-access—Have the interface operate in tagged-access mode. In this mode, the interface can be in multiple VLANs. Tagged access interfaces typically connect to network devices, such as PCs, printers, IP telephones, and IP cameras.</p> <p>trunk—Have the interface operate in trunk mode. In this mode, the interface can be in multiple VLANs and can multiplex traffic between different VLANs. Trunk interfaces typically connect to other switches and to routers on the LAN.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring Reflective Relay</i>• <i>Example: Configuring Reflective Relay for Use with VEPA Technology</i>• autostate-exclude on page 273

private-vlan

Syntax	<code>private-vlan (isolated community) vlan-id <i>number</i>;</code>
Hierarchy Level	<code>[edit vlans <i>vlan-name</i>]</code>
Release Information	Statement introduced in Junos OS Release 14.1X53-D10 for EX Series switches. Statement introduced in Junos OS Release 14.1X53-D15 for QFX Series switches.
Description	Configure a secondary VLAN (either an isolated VLAN or a community VLAN) within a private VLAN (PVLAN) and specify a VLAN ID for that secondary VLAN. This statement essentially converts a VLAN into a PVLAN, by carving out discrete subdomains (secondary VLANs) within the primary VLAN. You must specify a VLAN ID for each secondary PVLAN.
<div>  NOTE: After you have configured the secondary VLAN, you must also configure its association with a specific primary VLAN. See isolated-vlan and community-vlan for additional information. </div>	
Options	<ul style="list-style-type: none"> • isolated — The VLAN specified by <i>vlan-name</i> is defined as an <i>isolated</i> VLAN and a VLAN-ID is assigned to it. An isolated VLAN receives packets only from the primary VLAN and forwards frames upstream to the primary VLAN. The VLAN name is optional. The VLAN ID is required. • community — The VLAN specified by <i>vlan-name</i> is defined as community VLAN and a VLAN-ID is assigned to it. A <i>community</i> VLAN used to transport frames among members of a community, which is a subset of users within the VLAN, and to forward frames upstream to the primary VLAN. The VLAN name is optional. The VLAN ID is required.
Required Privilege Level	system—To view this statement in the configuration. system—control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Creating a Private VLAN on a Single Switch (CLI Procedure) on page 214 • Creating a Private VLAN Spanning Multiple Switches (CLI Procedure) on page 216

service-id

Syntax	<code>service-id <i>number</i>;</code>
Hierarchy Level	[edit switch-options] [edit vlans <i>vlan-name</i>]
Release Information	Statement introduced in Junos OS Release 12.3R2 for EX Series switches and MX Series routers. Statement introduced in Junos OS Release 13.2 for the QFX Series.
Description	Specify a service identifier for each multichassis aggregated Ethernet interface that belongs to a link aggregation group (LAG).
Options	number —A number that identifies a particular service. Range: 1 through 65535
Required Privilege Level	system —To view this statement in the configuration. system control —To add this statement to the configuration.

switch-options

Syntax	<pre> switch-options { interface <i>interface-name</i> { interface-mac-limit <i>limit</i> { packet-action drop; } no-mac-learning; static-mac <i>static-mac-address</i> { vlan-id <i>number</i>; } } interface-mac-limit <i>limit</i> { packet-action drop; } mac-statistics; mac-table-size <i>limit</i> { packet-action drop; } no-mac-learning; route-distinguisher service-id <i>number</i>; vrf-target vrf-target auto vrf-target import target vrf-target export target vrf-import vrf-export vtep-source-interface } </pre>
Hierarchy Level	<pre> [edit <i>number</i>], [edit vlans <i>vlan--name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> vlans <i>vlan-name</i>], [edit routing-instances <i>routing-instance-name</i> vlans <i>vlan-name</i>] </pre>
Release Information	<p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches and MX Series routers.</p> <p>Statement introduced in Junos OS Release 13.2 for the QFX Series.</p> <p>Statement options vrf-target target, vrf-target auto, vrf-target import target, vrf-target export target, vrf-import, and vrf-export added in Junos OS Release 14.1X53-D30 for EVPN VXLAN on QFX5100 switch.</p>
Description	<p>Configure Layer 2 learning and forwarding properties for a VLAN or a virtual switch.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>


static (Static MAC-Based VLANs)

Syntax	<pre>static { vlan <i>vlan-name</i> { mac <i>mac-address</i> { next-hop <i>interface-name</i>; } } }</pre>
Hierarchy Level	[edit ethernet-switching-options]
Release Information	Statement introduced in Junos OS Release 11.1 for EX Series switches.
Description	<p>Specify VLAN and MAC addresses to add to the Ethernet switching table.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure)



static-mac

Syntax	<pre>static-mac <i>mac-address</i>;</pre>
Hierarchy Level	[edit vlans <i>vlan-name</i> switch-options interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 13.2 for EX Series switches. Statement introduced in Junos OS Release 13.2 for the QFX Series.
Description	Specify a static MAC address to assign to this interface.
Options	<i>mac-address</i> —MAC address
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Adding a Static MAC Address Entry to the Ethernet Switching Table (CLI Procedure) on page 81

vlan-id (VLANs)

Syntax	<code>vlan-id <i>number</i>;</code>
Hierarchy Level	<p>For platforms without ELS:</p> <pre>[edit vlans <i>vlan-name</i> vlan-range]</pre> <p>For platforms without ELS and with ELS:</p> <pre>[edit vlans <i>vlan-name</i>]</pre> <p>For ELS platforms only:</p> <pre>[edit interfaces <i>interface-name</i> unit <i>number</i>] [edit vlans <i>vlan-name</i> vlan-id-list]</pre>
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	Configure an 802.1Q tag to apply to all traffic that originates on the VLAN.
Default	<p>On a QFX3500 and QFX3500 switch, if you use the default factory configuration, all traffic originating on the VLAN is untagged and has a VLAN identifier of 1. The number zero is reserved for priority tagging and the number 4093 is also reserved.</p> <p>On a QFX5100 switch, if you use the default factory configuration, all traffic originating on the VLAN is untagged and has a VLAN identifier of 1. The number zero is reserved for priority tagging and the number 4093 is also reserved.</p>
	<div>  <p>NOTE: You can only create up to 4090 VLANs on a QFX5100 switch. If you create more than 4090 VLANs, the interfaces associated with the extra VLANs are not displayed in the <code>show vlans</code> command output. For example, if you create 4094 VLANs, the extra VLANs will not have interfaces associated with the VLANs. The order in which you configure the extra VLANs determines which interfaces are missing from the <code>show vlans</code> command output.</p> </div>
Options	<p><i>number</i> —VLAN tag identifier.</p> <p>Range: 0 through 4093.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Example: Setting Up Bridging with Multiple VLANs</i> • <i>Understanding Bridging and VLANs</i>

vlan-id-list

Syntax	<code>vlan-id-list [<i>vlan-id-numbers</i>];</code>
Hierarchy Level	<p>[edit bridge-domains <i>bridge-domain-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]</p> <p>[edit interfaces <i>interface-name</i> unit 0],</p> <p>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>],</p> <p>[edit vlans <i>vlan-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 9.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Statement introduced in Junos OS Release 13.2 for the QFX Series.</p>
Description	<p>Specify a VLAN identifier list to use for a bridge domain or VLAN in trunk mode.</p> <p>Specify the trunk option in the interface-mode statement to accept packets with a VLAN ID that matches the list of VLAN IDs specified in the vlan-id-list statement to forward the packet within the bridge domain or VLAN configured with the matching VLAN ID. Specify the access option to accept packets with no VLAN ID to forward the packet within the bridge domain or VLAN configured with the VLAN ID that matches the VLAN ID specified in the vlan-id statement.</p> <p>This statement also enables you to bind a logical interface to a list of VLAN IDs, thereby configuring the logical interface to receive and forward a frame with a tag that matches the specified VLAN ID list.</p>
<div>  <p>WARNING: On some EX and QFX Series switches, you can apply no more than eight VLAN identifier lists to a physical interface.</p> </div>	
Options	<p><i>vlan-id-numbers</i>—Valid VLAN identifiers. You can combine individual numbers with range lists including a hyphen.</p> <p>Range: 0 through 4095</p>
<div>  <p>NOTE: On EX Series switches and the QFX Series, the range is 0 through 4094.</p> </div>	
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>

- Related Documentation**
- *Configuring a Bridge Domain*
 - *Configuring a VLAN*
 - *Configuring VLANs for EX Series Switches (CLI Procedure)*
 - *Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances*
 - *Configuring VLAN Identifiers for VLANs and VPLS Routing Instances*
 - *Configuring Q-in-Q Tunneling (CLI Procedure)*

vlan-rewrite

Syntax	<code>vlan-rewrite translate (200 500 201 501)</code>
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>number</i> family bridge interface-mode trunk] [edit interfaces <i>interface-name</i> unit <i>number</i> family ethernet-switching interface-mode trunk]
Release Information	Statement introduced in Junos OS Release 9.4. Statement introduced in Junos OS Release 12.3R2 for EX Series switches. Statement introduced in Junos OS Release 13.2 for the QFX Series.
Description	Translates an incoming VLAN to a bridge-domain VLAN, corresponding counter translation at egress. Supports translation of VLAN 200 to VLAN 500 and VLAN 201 to VLAN 501. Other valid VLANs pass through without translation.
Options	translate 200 500 —Translates incoming packets with VLAN 200 to 500. translate 201 501 —Translates incoming packets with VLAN 201 to 501. translate 202 502 —Translates incoming packets with VLAN 202 to 502.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>Rewriting a VLAN Tag and Adding a New Tag</i>

vlan-tagging

Syntax	vlan-tagging;
Hierarchy Level	[edit interfaces <i>interface-name</i>] [edit interfaces interface-range <i>interface-range-name</i>]
Release Information	Statement introduced in Junos OS Release 11.3 for the QFX Series.
Description	Enable VLAN tagging. The platform receives and forwards single-tag frames with 802.1Q VLAN tags.
Default	VLAN tagging is disabled by default.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>vlan-id</i>• <i>Configuring a Layer 3 Logical Interface</i>

vlan-tags

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

vlan

```

Syntax  vlans {
        vlan-name {
            description text-description;
            domain-type bridge;
            forwarding-options {
                dhcp-security {
                    arp-inspection;
                    group group-name {
                        interface interface-name {
                            static-ip ip-address {
                                mac mac-address;
                            }
                        }
                    }
                    overrides {
                        no-option82;
                        trusted;
                        untrusted;
                    }
                }
            }
            ip-source-guard;
            no-dhcp-snooping;
            option-82 {
                circuit-id {
                    prefix {
                        host-name;
                        logical-system-name;
                        routing-instance-name;
                    }
                    use-interface-description (device | logical);
                    use-vlan-id;
                }
                remote-id {
                    host-name hostname;
                    use-interface-description (device | logical);
                    use-string string;
                }
                vendor-id {
                    use-string string;
                }
            }
        }
    }
    fip-security {
        examine-vn2vf;
        examine-vn2vn {
            beacon-period milliseconds;
        }
        fc-map fc-map-value;
        interface interface-name {
            (fcoe-trusted | no-fcoe-trusted;)
        }
    }
}

```

```

l3-interface irb.logical-unit-number;
multicast-snooping-options {
  flood-groups [group-names];
  forwarding-cache {
    threshold {
      reuse threshold;
      suppress threshold;
    }
  }
  graceful-restart {
    disable;
    restart-duration duration;
  }
  host-outbound-traffic {
    dot1p bits;
    forwarding-class forwarding-class;
  }
  multichassis-lag-replicate-state;
  nexthop-hold-time time;
  options {
    syslog {
      level level;
      mark interval;
      upto level;
    }
  }
  traceoptions {
    file filename {
      files number;
      no-world-readable;
      size file-size;
      world-readable;
    }
    flag flag {
      disable;
    }
  }
}
switch-options {
  interface interface-name {
    interface-mac-limit limit {
      packet-action action;
    }
    static-mac mac-address;
  }
  interface-mac-limit limit {
    packet-action action;
  }
  mac-move-limit limit {
    packet-action action;
  }
  mac-table-size limit {
    packet-action drop;
  }
  no-mac-learning;
}

```

```
}
vlan-id number;
vlan-id-list [vlan-id | vlan-id-vlan-id];
vxlan {
    encapsulate-inner-vlan
    ingress-node-replication
    multicast-group
    ovsdb-managed
    unreachable-vtep-aging-timer
    vni
}
}
```

Hierarchy Level [edit]

Release Information Statement introduced in Junos OS Release 13.2 for the QFX Series.
Statements for private VLANs and Q-in-Q tunneling introduced in Junos OS Release 12.1 for the QFX Series.

Description Configure VLAN properties on the QFX Series.

Default If you use the default factory configuration, all switch interfaces become part of the VLAN default.

Options *vlan-name*—Name of the VLAN. The name can contain letters, numbers, hyphens (-), and periods (.) and can be up to 255 characters long.

The remaining statements are described separately.

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

Related Documentation

- [Understanding Bridging and VLANs on page 35](#)
- [Configuring VLANs on page 66](#)

CHAPTER 12

MAC Address Configuration Statements

- [mac-limit on page 315](#)
- [mac-notification on page 316](#)
- [mac-statistics on page 317](#)
- [mac-table-aging-time on page 318](#)
- [mac-table-size on page 319](#)
- [notification-interval on page 320](#)

mac-limit

Syntax	<code>mac-limit <i>number</i>;</code>
Hierarchy Level	[edit vlans <i>vlan-name</i>]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	Configure the number of MAC addresses allowed on a VLAN.
Default	MAC limit is disabled.
Options	<i>number</i> —Maximum number of MAC addresses. Range: 1 through 32768



NOTE: This statement is not supported on QFabric systems.

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

- | | |
|------------------------------|--|
| Related Documentation | <ul style="list-style-type: none">• show vlans on page 396• Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43• Configuring MAC Table Aging• Understanding Bridging and VLANs |
|------------------------------|--|

mac-notification

Syntax	<pre>mac-notification { notification-interval <i>seconds</i>; }</pre>
Hierarchy Level	[edit ethernet-switching-options] [edit switch-options]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series. Hierarchy level [edit switch-options] added in Junos OS Release 14.1X53-D10 for EX Series and QFX Series.
Description	<p>Enable MAC notification for a switch. If you configure this statement without setting a notification interval, MAC notification is enabled with the default MAC notification interval of 30 seconds.</p> <p>The remaining statement is explained separately.</p>
Default	MAC notification is disabled by default.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring MAC Notification</i>• Configuring MAC Notification (CLI Procedure) on page 92

mac-statistics

Syntax	mac-statistics;
Hierarchy Level	<p>[edit bridge-domains <i>bridge-domain-name</i> bridge-options],</p> <p>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> switch-options],</p> <p>[edit logical-systems <i>logical-system-name</i> switch-options],</p> <p>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options],</p> <p>[edit routing-instances <i>routing-instance-name</i> switch-options],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols evpn],</p> <p>[edit switch-options],</p> <p>[edit switch-options],</p> <p>[edit vlans <i>vlan-name</i> switch-options]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for the switch-options statement added in Junos OS Release 9.2.</p> <p>Support for top-level configuration for the virtual-switch type of routing instance added in Junos OS Release 9.2. In Junos OS Release 9.1 and earlier, the routing instances hierarchy supported this statement only for a VPLS instance or a bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>[edit switch-options] and [edit vlans <i>vlan-name</i> switch-options] hierarchy levels introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Support for EVPNs added in Junos OS Release 13.2 for MX 3D Series routers.</p> <p>[edit switch-options] and [edit vlans <i>vlan-name</i> switch-options] hierarchy levels introduced in Junos OS Release 13.2 for the QFX Series.</p>
Description	(MX Series routers, EX Series switches, and QFX Series only) For bridge domains or VLANs, enable MAC accounting either for a specific bridge domain or VLAN, or for a set of bridge domains or VLANs associated with a Layer 2 trunk port.
Default	disabled
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Layer 2 Learning and Forwarding for Bridge Domains Overview</i> • Layer 2 Learning and Forwarding for VLANs Overview on page 34 • <i>Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports</i> • <i>Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port</i> • <i>Configuring EVPN Routing Instances</i>

mac-table-aging-time

Syntax	<code>mac-table-aging-time seconds;</code>
Hierarchy Level	For platforms without ELS: [edit ethernet-switching-options], [edit vlan <i>vlan-name</i>] For platforms with ELS: [edit vlan <i>vlan-name</i> switch-options]
Release Information	Statement introduced for specific VLANs in Junos OS Release 11.1 for the QFX Series.
Description	Define how long entries remain in the Ethernet switching table before expiring: <ul style="list-style-type: none">• If you specify this statement at the [ethernet-switching-options] hierarchy level, it applies to all VLANs on the switch.• If you specify this statement at the [vlan] hierarchy level, it applies to the specified VLAN.
Default	300 seconds
Options	seconds —Time that entries remain in the Ethernet switching table before being removed. <ul style="list-style-type: none">• Range—60 to 1,000,000 seconds.• Default—300 seconds.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43• Configuring MAC Table Aging• Configuring MAC Table Aging on page 96• Understanding Bridging and VLANs on page 35• show ethernet-switching statistics aging on page 410

mac-table-size

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



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

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

Options	limit —Specify the maximum number of addresses in the MAC address table. Range: 16 through 1,048,575 MAC addresses Default: 5120 MAC addresses There is no default MAC address limit for the mac-table-size statement at the [edit switch-options] hierarchy level. The number of MAC addresses that can be learned is only limited by the platform, 65,535 MAC addresses for EX Series switches and 1,048,575 MAC addresses for other devices. The remaining statement is explained separately.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Layer 2 Learning and Forwarding for Bridge Domains Overview</i>• Layer 2 Learning and Forwarding for VLANs Overview on page 34• <i>Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports</i>• <i>Layer 2 Learning and Forwarding for VLANs Acting as a Switch for a Layer 2 Trunk Port</i>

notification-interval

Syntax	notification-interval <i>seconds</i> ;
Hierarchy Level	[edit ethernet-switching-options mac-notification] [edit switch-options mac-notification]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series. Hierarchy level [edit switch-options] added in Junos OS Release 14.1X53-D10 for EX Series and QFX Series.
Description	Configure the MAC notification interval for a switch. The MAC notification interval is the amount of time the switch waits before sending learned or unlearned MAC address SNMP notifications to the network management server. For instance, if the MAC notification interval is set to 10, all of the MAC address addition and removal SNMP notifications are sent to the network management system every 10 seconds.
Options	seconds —The MAC notification interval, in seconds. Range: 1 through 60 Default: 30
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring MAC Notification</i>• Configuring MAC Notification (CLI Procedure) on page 92

CHAPTER 13

Private VLAN Configuration Statements

- [extend-secondary-vlan-id](#) on page 321
- [isolated](#) on page 322
- [isolation-vlan-id](#) on page 322
- [primary-vlan](#) on page 323
- [promiscuous](#) on page 323
- [vlans](#) on page 324

[extend-secondary-vlan-id](#)

Syntax	<code>extend-secondary-vlan-id <i>number</i>;</code>
Hierarchy Level	[edit vlans <i>vlan-name</i> pvlan]
Release Information	Statement introduced in Junos OS Release 12.2 for the QFX Series.
Description	Configure traffic that egresses from a secondary VLAN trunk port to retain its secondary VLAN tag instead of getting the tag of the primary VLAN that the secondary port is a member of.
Required Privilege Level	routing—To view this statement in the configuration. routing—control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Understanding Secondary VLAN Trunk Ports and Promiscuous Access Ports on PVLANS</i>• <i>Example: Configuring PVLANS with Secondary VLAN Trunk Ports and Promiscuous Access Ports</i>


isolated

Syntax	isolated;
Hierarchy Level	[edit vlan <i>vlan-name</i> interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Configure an access or trunk port to be isolated. You configure a trunk port to be isolated so that it can be a secondary VLAN trunk port—that is, it can carry secondary VLAN traffic.
Required Privilege Level	system—To view this statement in the configuration. system-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Creating a Private VLAN on a Single Switch</i>• <i>Creating a Private VLAN Spanning Multiple Switches</i>• <i>Understanding Secondary VLAN Trunk Ports and Promiscuous Access Ports on PVLANS</i>• <i>Example: Configuring PVLANS with Secondary VLAN Trunk Ports and Promiscuous Access Ports</i>

isolation-vlan-id

Syntax	isolation-vlan-id <i>number</i> ;
Hierarchy Level	[edit vlan <i>vlan-name</i> pvlan]
Release Information	Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Configure an interswitch isolated VLAN within a private VLAN that spans multiple switches.
Options	<i>number</i> —VLAN tag identifier. Range: 0 through 4093
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Creating a Private VLAN on a Single Switch</i>• <i>Creating a Private VLAN Spanning Multiple Switches</i>

primary-vlan

Syntax	<code>primary-vlan <i>vlan-name</i>;</code>
Hierarchy Level	[edit vlan <i>vlan-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	<p>Configure the primary VLAN for this community VLAN. The primary VLAN must be tagged, and the community VLAN must be untagged.</p> <p>If you want to create a community VLAN, you must configure the primary VLAN to be private using the <i>pvlan</i> statement.</p>
	<div>  <p>TIP: To display a list of all configured VLANs on the system, including VLANs that are configured but not committed, type ? after <i>vlan</i> or <i>vlan</i>s in your configuration mode command line. Note that only one VLAN is displayed for a VLAN range.</p> </div>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Creating a Private VLAN on a Single Switch</i> • <i>Creating a Private VLAN Spanning Multiple Switches</i>

promiscuous

Syntax	<code>promiscuous;</code>
Hierarchy Level	[edit vlan <i>vlan-name</i> interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 12.1 for the QFX Series.
Description	Configure an access or trunk port to be promiscuous.
Required Privilege Level	<p>system—To view this statement in the configuration.</p> <p>system-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Creating a Private VLAN on a Single Switch</i> • <i>Creating a Private VLAN Spanning Multiple Switches</i>

vlan

Syntax	<pre> vlan { vlan-name { description text-description; dot1q-tunneling { customer-vlans (id range); } filter input filter-name; filter output filter-name; interface interface-name { isolated; mapping (policy tag push native push); promiscuous; } isolation-vlan-id; l3-interface vlan.logical-interface-number; mac-limit number; mac-table-aging-time seconds; no-local-switching; no-mac-learning; primary-vlan vlan-name; pvlan extend-secondary-vlan-id vlan-id; vlan-id number; vlan-range vlan-id-low-vlan-id-high; } } </pre>
Hierarchy Level	[edit]
Release Information	<p>Statement introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>Statements for private VLANs and Q-in-Q tunneling introduced in Junos OS Release 12.1 for the QFX Series.</p>
Description	Configure VLAN properties on the QFX Series.
Default	If you use the default factory configuration, all switch interfaces become part of the VLAN default.
Options	<p>vlan-name—Name of the VLAN. The name can contain letters, numbers, hyphens (-), and periods (.) and can be up to 255 characters long.</p> <p>The remaining statements are described separately.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> Configuring VLANs Configuring Q-in-Q Tunneling Creating a Series of Tagged VLANs

- [Configuring IRB Interfaces on page 72](#)
- *Creating a Private VLAN on a Single Switch*
- *Understanding Bridging and VLANs*

STP Configuration Statements

- [bpdu-block on page 328](#)
- [bpdu-block-on-edge on page 329](#)
- [bpdu-timeout-action on page 330](#)
- [configuration-name on page 331](#)
- [cost on page 332](#)
- [bpdu-block on page 333](#)
- [bpdu-block-on-edge on page 333](#)
- [bpdu-timeout-action on page 334](#)
- [configuration-name on page 335](#)
- [cost on page 336](#)
- [disable-timeout \(Spanning Trees\) on page 337](#)
- [edge on page 338](#)
- [force-version \(IEEE 802.1D STP\) on page 339](#)
- [forward-delay on page 340](#)
- [hello-time on page 341](#)
- [interface \(BPDU\) on page 342](#)
- [max-age on page 343](#)
- [max-hops on page 344](#)
- [mode \(Protocols STP\) on page 345](#)
- [mstp on page 346](#)
- [no-root-port on page 347](#)
- [priority \(Protocols STP\) on page 348](#)
- [protocol on page 349](#)
- [protocols \(STP Type\) on page 350](#)
- [revision-level on page 351](#)
- [rstp on page 352](#)
- [traceoptions \(Spanning Tree\) on page 353](#)
- [vlan \(VSTP\) on page 356](#)

- [vlan-group on page 357](#)
- [vstp on page 358](#)

bpdu-block

Syntax	<pre>bpdu-block { interface (<i>interface-name</i> disable all); disable-timeout <i>seconds</i>; }</pre>
Hierarchy Level	[edit protocols layer2-control]
Release Information	Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.
Description	<p>Enable BPDU blocking on an interface.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• <i>BPDU Protection for Spanning-Tree Instance Interfaces Overview</i>• <i>BPDU Protection for Individual Spanning-Tree Instance Interfaces</i>• <i>Configuring BPDU Protection on Individual Interfaces</i>• <i>show spanning-tree bridge</i>• <i>show spanning-tree interface</i>• <i>Understanding BPDU Protection for STP, RSTP, and MSTP on EX Series Switches</i>

bpdu-block-on-edge

Syntax	bpdu-block-on-edge;
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)], [edit protocols (mstp rstp vstp)], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)]
Release Information	Statement introduced in Junos OS Release 9.4. Support for logical systems added in Junos OS Release 9.6.
Description	Enable BPDU blocking on the edge ports of a virtual switch.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>BPDU Protection for Spanning-Tree Instance Interfaces Overview</i> • <i>BPDU Protection on All Edge Ports of the Bridge</i> • <i>Configuring BPDU Protection on All Edge Ports</i>

bpdu-timeout-action

Syntax	bpdu-timeout-action (log block);
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)], [edit protocols (mstp rstp vstp)], [edit protocols (mstp rstp vstp) interface], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)]
Release Information	Statement introduced in Junos OS Release 9.4. Support for logical systems added in Junos OS Release 9.6.
Description	Provide STP loop protection for a given STP family protocol interface.
Default	If the bpdu-timeout-action statement is not configured, an interface that stops receiving BPDUs will transition to the designated port (forwarding) state, creating a potential loop.
Options	log —The interface logs the fact that it has not received BPDUs during the timeout interval. block —The interface is blocked and the fact that the interface has not received BPDUs during the timeout interval is logged.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Loop Protection for Spanning-Tree Instance Interfaces Overview</i>• <i>Configuring Loop Protection for a Spanning-Tree Instance Interface</i>• <i>Example: Enabling Loop Protection for Spanning-Tree Protocols</i>

configuration-name

Syntax	<code>configuration-name <i>configuration-name</i>;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mstp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp], [edit protocols mstp], [edit routing-instances <i>routing-instance-name</i> protocols mstp]
Release Information	Statement introduced in Junos OS Release 8.4. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for logical systems added in Junos OS Release 9.6.
Description	Specify the configuration name , which is the MSTP region name carried in the MSTP BPDUs.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>BPDU Overview</i> • <i>Configuring Multiple Spanning-Tree Protocol</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123 • <i>Understanding MSTP for EX Series Switches</i>

cost

Syntax	<code>cost cost;</code>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Configure link cost to control which bridge is the designated bridge and which port is the designated port. By default, the link cost is determined by the link speed.
Options	<p>cost—(Optional) Link cost associated with the port.</p> <p>Range: 1 through 200,000,000</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Spanning-Tree Instance Interface</i> • <i>Spanning-Tree Instance Interface Cost</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • <i>Understanding RSTP for EX Series Switches</i> • <i>Understanding MSTP for EX Series Switches</i> • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

bpdu-block

Syntax	bpdu-block { interface (<i>interface-name</i> disable all); disable-timeout <i>seconds</i> ; }
Hierarchy Level	[edit protocols layer2-control]
Release Information	Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.
Description	Enable BPDU blocking on an interface. The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>BPDU Protection for Spanning-Tree Instance Interfaces Overview</i> • <i>BPDU Protection for Individual Spanning-Tree Instance Interfaces</i> • <i>Configuring BPDU Protection on Individual Interfaces</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • <i>Understanding BPDU Protection for STP, RSTP, and MSTP on EX Series Switches</i>

bpdu-block-on-edge

Syntax	bpdu-block-on-edge;
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)], [edit protocols (mstp rstp vstp)], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)]
Release Information	Statement introduced in Junos OS Release 9.4. Support for logical systems added in Junos OS Release 9.6.
Description	Enable BPDU blocking on the edge ports of a virtual switch.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>BPDU Protection for Spanning-Tree Instance Interfaces Overview</i> • <i>BPDU Protection on All Edge Ports of the Bridge</i> • <i>Configuring BPDU Protection on All Edge Ports</i>

bpdu-timeout-action

Syntax	bpdu-timeout-action (log block);
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)], [edit protocols (mstp rstp vstp)], [edit protocols (mstp rstp vstp) interface], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)]
Release Information	Statement introduced in Junos OS Release 9.4. Support for logical systems added in Junos OS Release 9.6.
Description	Provide STP loop protection for a given STP family protocol interface.
Default	If the bpdu-timeout-action statement is not configured, an interface that stops receiving BPDUs will transition to the designated port (forwarding) state, creating a potential loop.
Options	log —The interface logs the fact that it has not received BPDUs during the timeout interval. block —The interface is blocked and the fact that the interface has not received BPDUs during the timeout interval is logged.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Loop Protection for Spanning-Tree Instance Interfaces Overview</i>• <i>Configuring Loop Protection for a Spanning-Tree Instance Interface</i>• <i>Example: Enabling Loop Protection for Spanning-Tree Protocols</i>

configuration-name

Syntax	<code>configuration-name <i>configuration-name</i>;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mstp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp], [edit protocols mstp], [edit routing-instances <i>routing-instance-name</i> protocols mstp]
Release Information	Statement introduced in Junos OS Release 8.4. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for logical systems added in Junos OS Release 9.6.
Description	Specify the configuration name , which is the MSTP region name carried in the MSTP BPDUs.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>BPDU Overview</i> • <i>Configuring Multiple Spanning-Tree Protocol</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123 • <i>Understanding MSTP for EX Series Switches</i>

cost

Syntax	<code>cost cost;</code>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Configure link cost to control which bridge is the designated bridge and which port is the designated port. By default, the link cost is determined by the link speed.
Options	<p>cost—(Optional) Link cost associated with the port.</p> <p>Range: 1 through 200,000,000</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Spanning-Tree Instance Interface</i> • <i>Spanning-Tree Instance Interface Cost</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • <i>Understanding RSTP for EX Series Switches</i> • <i>Understanding MSTP for EX Series Switches</i> • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

disable-timeout (Spanning Trees)

Syntax	<code>disable-timeout <i>seconds</i>;</code>
Hierarchy Level	[edit protocols layer2-control bpdu-block]
Release Information	Statement introduced in Junos OS Release 9.1 for EX Series switches.
Description	For interfaces configured for BPDU protection, specify the amount of time an interface is disabled by BPDU blocking. If this option is not configured, the interface is not periodically checked and remains disabled.
Default	The disable timeout is not enabled.
Options	<i>seconds</i> —Amount of time, in seconds, the interface receiving BPDUs protect is disabled. The range is 10 through 3600 seconds.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • <i>Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring BPDU Protection on Edge Interfaces to Prevent STP Miscalculations on EX Series Switches on page 147 • <i>Understanding BPDU Protection for STP, RSTP, and MSTP on EX Series Switches</i>

edge

Syntax	edge;
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Configure interfaces as edge ports. Edge ports do not expect to receive BPDUs. If a BPDU is received, the port becomes a nonedge port.
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Spanning-Tree Instance Interface</i> • <i>Spanning-Tree Instance Interface Configured as an Edge Port</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123 • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

force-version (IEEE 802.1D STP)

Syntax	force-version stp;
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (rstp vstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (rstp vstp)], [edit protocols (rstp vstp)], [edit routing-instances <i>routing-instance-name</i> protocols (rstp vstp)]
Release Information	Statement introduced in Junos OS Release 8.4. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for logical systems added in Junos OS Release 9.6.
Description	Force the spanning-tree protocol version to be the original IEEE 802.1D STP.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>Spanning-Tree Protocols Supported</i> • <i>RSTP or VSTP Forced to Run as IEEE 802.1D STP</i> • <i>Forcing RSTP or VSTP to Run as IEEE 802.1D STP (CLI Procedure)</i> • <i>Reverting to RSTP or VSTP from Forced IEEE 802.1D STP</i> • <i>Understanding RSTP for EX Series Switches</i> • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

forward-delay

Syntax	<code>forward-delay seconds;</code>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols vstp <i>vlan vlan-id</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp)],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan vlan-id</i>],</p> <p>[edit protocols (mstp rstp)],</p> <p>[edit protocols vstp <i>vlan vlan-id</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan vlan-id</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	For Rapid Spanning Tree Protocol (RSTP), VLAN Spanning Tree Protocol (VSTP), or Multiple Spanning Tree Protocol (MSTP), specify how long a bridge interface remains in the listening and learning states before transitioning to the forwarding state.
Options	<p>seconds—(Optional) Number of seconds the bridge port remains in the listening and learning states.</p> <p>Range: 4 through 30</p> <p>Default: 15 seconds</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Forward Delay Before Ports Transition to Forwarding State</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123 • <i>Understanding MSTP for EX Series Switches</i> • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

hello-time

Syntax	<code>hello-time seconds;</code>
Hierarchy Level	<pre>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> protocols vstp <i>vlan</i> <i>vlan-id</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan</i> <i>vlan-id</i>], [edit protocols (mstp rstp)], [edit protocols vstp <i>vlan</i> <i>vlan-id</i>], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)], [edit routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan</i> <i>vlan-id</i>]</pre>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Specify the number of seconds between transmissions of configuration BPDUs by the root bridge.
Options	<p>seconds—(Optional) Number of seconds between transmissions of configuration BPDUs.</p> <p>Range: 1 through 10</p> <p>Default: 2 seconds</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Hello Time for Root Bridge to Transmit Hello BPDUs</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123 • <i>Understanding MSTP for EX Series Switches</i> • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

interface (BPDU)

Syntax	interface (all [<i>interface-name</i>]) { drop; }
Hierarchy Level	<ul style="list-style-type: none">• For platforms with ELS CLI: [edit protocols layer2-control]• For platforms with Original CLI: [edit ethernet-switching-options]
Release Information	Statement introduced in Junos OS Release 11.1 for the QFX Series.
Description	Apply BPDU protection to all interfaces or one or more interfaces.
Options	<p>all—All interfaces.</p> <p><i>interface-name</i>—Name of the interface.</p> <p>drop—Drops xSTP BPDUs.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none">• <i>Example: Configuring Network Regions for VLANs with MSTP</i>• <i>Example: Configuring Faster Convergence and Improving Network Stability with RSTP</i>• Understanding BPDU Protection for STP, RSTP, and MSTP on page 146• show spanning-tree bridge on page 419• show spanning-tree interface on page 424

max-age

Syntax	<code>max-age seconds;</code>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> protocols vstp <i>vlan vlan-id</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan vlan-id</i>], [edit protocols (mstp rstp)], [edit protocols vstp <i>vlan vlan-id</i>], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)], [edit routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan vlan-id</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Specify the maximum expected arrival time of hello BPDUs.
Options	<p>seconds—(Optional) Number of seconds expected between hello BPDUs.</p> <p>Range: 6 through 40</p> <p>Default: 20 seconds</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Maximum Age for Awaiting Arrival of Hello BPDUs</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123 • <i>Understanding MSTP for EX Series Switches</i> • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

max-hops

Syntax	<code>max-hops hops;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mstp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp], [edit protocols mstp], [edit routing-instances <i>routing-instance-name</i> protocols mstp]
Release Information	Statement introduced in Junos OS Release 8.4. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for logical systems added in Junos OS Release 9.6.
Description	Configure the maximum number of hops a BPDU can be forwarded in the MSTP region.
Options	hops —(Optional) Number of hops the BPDU can be forwarded. Range: 1 through 255 Default: 19 hops
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring Multiple Spanning-Tree Protocol</i>• <i>show spanning-tree bridge</i>• <i>show spanning-tree interface</i>• Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101• <i>Understanding MSTP for EX Series Switches</i>

mode (Protocols STP)

Syntax	<code>mode (p2p shared);</code>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Configure link mode to identify point-to-point links.



NOTE: When the link is configured as full-duplex, the default link mode is p2p. When the link is configured half-duplex, the default link mode is shared.

Options	<p>p2p—The link is point to point.</p> <p>shared—The link is shared media.</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Spanning-Tree Instance Interface</i> • <i>Spanning-Tree Instance Interface Point-to-Point Link Mode</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • Example: Configuring Faster Convergence and Improved Network Stability with RSTP on EX Series Switches on page 123 • <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

mstp

```
Syntax  mstp {
    bpdu-block-on-edge;
    bridge-priority priority;
    configuration-name configuration-name;
    disable;
    forward-delay seconds;
    hello-time seconds;
    interface interface-name {
        bpdu-timeout-action {
            alarm;
            block;
        }
        cost cost;
        edge;
        mode (p2p | shared);
        no-root-port;
        priority interface-priority;
    }
    max-age seconds;
    max-hops hops;
    priority-hold-time seconds;
    revision-level revision-level;
    interface interface-name {
        bpdu-timeout-action {
            alarm;
            block;
        }
        cost cost;
        edge;
        mode (p2p | shared);
        no-root-port;
        priority interface-priority;
    }
    msti msti-id {
        bridge-priority priority;
        interface interface-name {
            cost cost;
            edge;
            priority interface-priority;
        }
        vlan vlan-id;
    }
    traceoptions {
        file filename <files number> <size size> <world-readable | no-world-readable>;
        flag flag <flag-modifier> <disable>;
    }
}
```

Hierarchy Level [edit protocols]

Release Information Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.

Description	Configure MSTP parameters.
Options	The 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	<ul style="list-style-type: none"> • <i>Configuring Multiple Spanning-Tree Protocol</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101

no-root-port

Syntax	no-root-port;
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols vstp <i>vlan</i> <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan</i> <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit protocols vstp <i>vlan</i> <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vstp <i>vlan</i> <i>vlan-id</i> interface <i>interface-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 9.1.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Ensure the port is the spanning-tree designated port. If the port receives superior bridge protocol data unit (BPDU) packets, root protect moves this port to a root-prevented spanning-tree state.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>Root Protection for Spanning-Tree Instance Interfaces Overview</i> • <i>Root Protect for a Spanning-Tree Instance Interface</i> • <i>Enabling Root Protect for a Spanning-Tree Instance Interface</i>

priority (Protocols STP)

Syntax	<code>priority interface-priority;</code>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit protocols vstp <i>vlan-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Use the interface priority to control which interface is elected as the root port. The interface priority must be set in increments of 16.
Options	<p>priority—(Optional) Interface priority.</p> <p>Range: 0 through 240</p>
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Spanning-Tree Instance Interface</i> • <i>Spanning-Tree Instance Interface Configured as an Edge Port</i> • <i>Spanning-Tree Instance Interface Priority</i> • <i>[edit protocols mstp] Configuration Statement Hierarchy on EX Series Switches</i> • <i>[edit protocols rstp] Configuration Statement Hierarchy on EX Series Switches</i> • <i>[edit protocols vstp] Configuration Statement Hierarchy on EX Series Switches</i>

protocol

Syntax	<code>protocol (cdp stp vtp pvstp);</code>
Hierarchy Level	[edit protocols layer2-control mac-rewrite interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 9.1. Statement introduced in Junos OS Release 13.2 for QFX series. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for PVSTP introduced in Junos OS Release 13.3.
Description	Configure the protocol to be tunneled on an interface for Layer 2 protocol tunneling. To tunnel multiple protocols, include multiple protocol statements.
Options	<p>cdp—Tunnel the Cisco discovery protocol.</p> <p>stp—Tunnel all versions of the spanning-tree protocol.</p> <p>vtp—Tunnel the VLAN trunk protocol.</p> <p>pvstp—Tunnel the Per-VLAN Spanning Tree Plus (PVST+) protocol</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Layer 2 Protocol Tunneling Through a Network Overview</i> • <i>Layer 2 Protocol Tunnel Interface</i> • <i>Layer 2 Protocol to be Tunneled</i> • <i>Configuring Layer 2 Protocol Tunneling</i>

protocols (STP Type)

Syntax	<pre>protocols { mstp { ... } rstp { ... } vstp { ... } }</pre>
Hierarchy Level	[edit], [edit logical-systems <i>logical-system-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i>], [edit routing-instances <i>routing-instance-name</i>]
Release Information	Statement introduced in Junos OS Release 8.4. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for logical systems added in Junos OS Release 9.6.
Description	Configure the Spanning Tree Protocol type as MSTP, RSTP, or VSTP.
Options	mstp —Configure the protocol as Multiple Spanning Tree. rstp —Configure the protocol as Rapid Spanning Tree. vstp —Configure the protocol as VLAN Spanning Tree.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring Rapid Spanning-Tree Protocol</i>• <i>Configuring Multiple Spanning-Tree Protocol</i>• <i>Configuring VLAN Spanning Tree Protocol</i>• <i>Understanding MSTP for EX Series Switches</i>

revision-level

Syntax	<code>revision-level <i>revision-level</i>;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mstp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp], [edit protocols mstp], [edit routing-instances <i>routing-instance-name</i> protocols mstp]
Release Information	Statement introduced in Junos OS Release 8.4. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for logical systems added in Junos OS Release 9.6.
Description	Set the revision number of the MSTP configuration.
Options	<i>revision-level</i> —Configure the revision number of the MSTP region configuration. Range: 0 through 65,535
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>Configuring Multiple Spanning-Tree Protocol</i> • <i>show spanning-tree bridge</i> • <i>show spanning-tree interface</i> • Example: Configuring Network Regions for VLANs with MSTP on EX Series Switches on page 101 • <i>Understanding MSTP for EX Series Switches</i>

rstp

Syntax	<pre> rstp { bpd-block-on-edge; bpd-destination-mac-address provider-bridge-group; bridge-priority <i>priority</i>; disable; extended-system-id; force-version stp; forward-delay <i>seconds</i>; hello-time <i>seconds</i>; interface <i>interface-name</i> { bpd-timeout-action { alarm; block; } cost <i>cost</i>; edge; mode (p2p shared); no-root-port; priority <i>interface-priority</i>; } max-age <i>seconds</i>; priority-hold-time <i>seconds</i>; traceoptions { file <i>filename</i> <files <i>number</i>> <size <i>size</i>> <world-readable no-world-readable>; flag <i>flag</i> <<i>flag-modifier</i>> <disable>; } } </pre>
Hierarchy Level	[edit protocols]
Release Information	Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.
Description	Configure RSTP parameters.
Options	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	<ul style="list-style-type: none"> Configuring RSTP (CLI Procedure) on page 139

traceoptions (Spanning Tree)

Syntax	<pre>traceoptions { file <i>filename</i> <files <i>number</i>> <size <i>size</i>> <world-readable no-world-readable>; flag <i>flag</i> <flag-modifier> <disable>; }</pre>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp)],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)],</p> <p>[edit protocols (mstp rstp vstp vstp vlan <i>vlan-id</i>)],</p> <p>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Set protocol-level tracing options for spanning-tree protocols..
Default	The default STP protocol-level trace options are inherited from the global traceoptions statement.
Options	<p>disable—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as all.</p> <p>file <i>filename</i>—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place STP tracing output in the file <code>/var/log/stp-log</code>.</p> <p>files <i>number</i>—(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.</p> <p>If you specify a maximum number of files, you must also specify a maximum file size with the size option.</p> <p>Range: 2 through 1000 files</p> <p>Default: 1 trace file only</p> <p>flag—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements. The following are the STP-specific tracing options:</p> <ul style="list-style-type: none"> • all—Trace all operations. • all-failures—Trace all failure conditions. • bpdud—Trace BPDU reception and transmission. • bridge-detection-state-machine—Trace the bridge detection state machine.

- **events**—Trace events of the protocol state machine.
- **port-information-state-machine**—Trace the port information state machine.
- **port-migration-state-machine**—Trace the port migration state machine.
- **port-receive-state-machine**—Trace the port receive state machine.
- **port-role-transit-state-machine**—Trace the port role transit state machine.
- **port-role-select-state-machine**—Trace the port role selection state machine.
- **port-state-transit-state-machine**—Trace the port state transit state machine.
- **port-transmit-state-machine**—Trace the port transmit state machine.
- **ppmd**—Trace the state and events for the ppm process.
- **state-machine-variables**—Trace when the state machine variables change.
- **timers**—Trace protocol timers.
- **topology-change-state-machine**—Trace the topology change state machine.

The following are the global tracing options:

- **all**—All tracing operations.
- **config-internal**—Trace configuration internals.
- **general**—Trace general events.
- **normal**—All normal events.

Default: If you do not specify this option, only unusual or abnormal operations are traced.

- **parse**—Trace configuration parsing.
- **policy**—Trace policy operations and actions.
- **regex-parse**—Trace regular-expression parsing.
- **route**—Trace routing table changes.
- **state**—Trace state transitions.
- **task**—Trace protocol task processing.
- **timer**—Trace protocol task timer processing.

no-world-readable—(Optional) Prevent any user from reading the log file.

size size—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). 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 must also specify a maximum number of trace files with the **files** option.

Syntax: **xk** to specify KB, **xm** to specify MB, or **xg** to specify GB

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	• <i>Spanning-Tree Protocol Trace Options</i>
	• <i>Tracing Spanning-Tree Operations</i>
	• <i>Example: Tracing Spanning-Tree Protocol Operations</i>

vlan (VSTP)

Syntax	<pre>vlan <i>vlan-id</i> { bridge-priority <i>priority</i>; forward-delay <i>seconds</i>; hello-time <i>seconds</i>; max-age <i>seconds</i>; interface <i>interface-name</i> { cost <i>cost</i>; edge; mode (p2p shared); no-root-port; priority <i>interface-priority</i>; } }</pre>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols vstp], [edit protocols vstp]
Release Information	Statement introduced in Junos OS Release 9.0. Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches. Support for logical systems added in Junos OS Release 9.6.
Description	Configure VSTP VLAN parameters.
Options	The 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	<ul style="list-style-type: none">• <i>Configuring VLAN Spanning Tree Protocol</i>• <i>Understanding VSTP for EX Series Switches and QFX Series Switches</i>

vlan-group

Syntax `vlan-group group group-name {
 interface (Spanning Trees) (all | interface-name) {
 arp-on-stp;
 traceoptions (Spanning Trees) {
 file filename <files number > <size size > <no-stamp | no-world-readable | world-readable>;
 flag flag;
 }
 }`

Hierarchy Level [edit protocols vstp]

Description Configure VLAN group for Spanning Tree Protocol (VSTP). VSTP is used to prevent loops in Layer 2 networks on a per-VLAN basis.



BEST PRACTICE: Configure RSTP when you configure VSTP. RSTP overhead is minimal and this configuration ensures that a spanning-tree protocol is running on all VLANs on your switch, even when your switch is supporting more than the maximum number of allowed VSTP VLANs.

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

- *vstp*
- *show spanning-tree bridge*
- *show spanning-tree interface*
- *Understanding VSTP for EX Series Switches and QFX Series Switches*

vstp

```
Syntax  vstp {
        bpdu-block-on-edge;
        disable;
        force-version stp;
        interface interface-name {
            bpdu-timeout-action {
                alarm;
                block;
            }
            cost cost;
            edge;
            mode (p2p | shared);
            no-root-port;
            priority interface-priority;
        }
        priority-hold-time seconds;
        vlan vlan-id {
            bridge-priority priority;
            forward-delay seconds;
            hello-time seconds;
            max-age seconds;
            interface interface-name {
                access-trunk
                bpdu-timeout-action {
                    alarm;
                    block;
                }
                cost cost;
                edge;
                mode (p2p | shared);
                no-root-port;
                priority interface-priority;
            }
        }
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
```

Hierarchy Level [edit protocols]

Release Information Statement introduced in Junos OS Release 13.2X50-D10 for EX Series switches.

Description Configure VSTP parameters.

Options The 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 VLAN Spanning Tree Protocol*

CHAPTER 15

Q-in-Q Configuration Statements

- [flexible-vlan-tagging on page 362](#)
- [input-vlan-map on page 363](#)
- [native-vlan-id on page 364](#)
- [output-vlan-map on page 365](#)
- [pop on page 366](#)
- [push on page 367](#)
- [swap on page 368](#)
- [vlan-id-list on page 369](#)

flexible-vlan-tagging

Syntax	flexible-vlan-tagging;
Hierarchy Level	[edit interfaces aex], [edit interfaces ge- <i>fpc/pic/port</i>], [edit interfaces et- <i>fpc/pic/port</i>], [edit interfaces ps0], [edit interfaces xe- <i>fpc/pic/port</i>]
Release Information	Statement introduced in Junos OS Release 8.1. Support for aggregated Ethernet added in Junos OS Release 9.0. Statement introduced in Junos OS Release 12.1x48 for PTX Series Packet Transport Routers. Statement introduced in Junos OS Release 13.2X50-D15 for EX Series switches. Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.
Description	Support simultaneous transmission of 802.1Q VLAN single-tag and dual-tag frames on logical interfaces on the same Ethernet port, and on pseudowire logical interfaces. This statement is supported on M Series and T Series routers, for Fast Ethernet and Gigabit Ethernet interfaces only on Gigabit Ethernet IQ2 and IQ2-E, IQ, and IQE PICs, and for aggregated Ethernet interfaces with member links in IQ2, IQ2-E, and IQ PICs or in MX Series DPCs, or on Ethernet interfaces for PTX Series Packet Transport Routers or 100-Gigabit Ethernet Type 5 PIC with CFP. This statement is supported on Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces on EX Series switches.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Configuring Mixed Tagging</i>• <i>Configuring Flexible VLAN Tagging on PTX Series Packet Transport Routers</i>

input-vlan-map

Syntax	<pre>input-vlan-map { (pop pop-pop pop-swap push push-push swap swap-push swap-swap); inner-tag-protocol-id <i>tpid</i>; inner-vlan-id <i>number</i>; tag-protocol-id <i>tpid</i>; vlan-id <i>number</i>; }</pre>
Hierarchy Level	<pre>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]</pre>
Release Information	<p>Statement introduced before Junos OS Release 7.4.</p> <p>pop-pop, pop-swap, push-push, swap-push, and swap-swap statements introduced in Junos OS Release 8.1.</p> <p>Statement introduced in Junos OS Release 13.2X50-D15 for EX Series switches.</p> <p>Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.</p>
Description	<p>For Gigabit Ethernet IQ, 10-Gigabit Ethernet SFPP interfaces, 100-Gigabit Ethernet Type 5 PIC with CFP only as well as Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces, define the rewrite profile to be applied to incoming frames on this logical interface.</p> <p>The statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Stacking a VLAN Tag</i> • output-vlan-map on page 365 • <i>Configuring Q-in-Q Tunneling (CLI Procedure)</i>

native-vlan-id


Syntax	<code>native-vlan-id <i>number</i>;</code>
Hierarchy Level	[edit interfaces <i>ge-fpc/pic/port</i>], [edit interfaces <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 8.3. Statement introduced in Junos OS Release 12.2 for ACX Series Universal Access Routers. Statement introduced in Junos OS Release 12.3R2 for EX Series switches. Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.
Description	<p>Configure mixed tagging support for untagged packets on a port for the following:</p> <ul style="list-style-type: none"> • M Series routers with Gigabit Ethernet IQ PICs with SFP and Gigabit Ethernet IQ2 PICs with SFP configured for 802.1Q flexible VLAN tagging • MX Series routers with Gigabit Ethernet DPCs and MICs, Tri-Rate Ethernet DPCs and MICs, and 10-Gigabit Ethernet DPCs and MICs and MPCs configured for 802.1Q flexible VLAN tagging • T4000 routers with 100-Gigabit Ethernet Type 5 PIC with CFP • EX Series switches with Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces <p>When the native-vlan-id statement is included with the flexible-vlan-tagging statement, untagged packets are accepted on the same mixed VLAN-tagged port.</p> <p>The logical interface on which untagged packets are received must be configured with the same VLAN ID as the native VLAN ID configured on the physical interface. To configure the logical interface, include the vlan-id statement (matching the native-vlan-id statement on the physical interface) at the [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>] hierarchy level.</p> <p>When the native-vlan-id statement is included with the interface-mode statement, untagged packets are accepted and forwarded within the bridge domain or VLAN that is configured with the matching VLAN ID.</p>
Options	<p><i>number</i>—VLAN ID number.</p> <p>Range: (ACX Series routers and EX Series switches) 0 through 4094.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Configuring Mixed Tagging Support for Untagged Packets</i> • <i>Configuring a Logical Interface for Access Mode</i> • Configuring the Native VLAN Identifier (CLI Procedure) on page 67 • <i>Understanding Bridging and VLANs on EX Series Switches</i>

- [flexible-vlan-tagging on page 362](#)
- *Understanding Q-in-Q Tunneling on EX Series Switches*


output-vlan-map

Syntax	<pre>output-vlan-map { (pop pop-pop pop-swap push push-push swap swap-push swap-swap); inner-tag-protocol-id <i>tpid</i>; inner-vlan-id <i>number</i>; tag-protocol-id <i>tpid</i>; vlan-id <i>number</i>; }</pre>
Hierarchy Level	<pre>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]</pre>
Release Information	<p>Statement introduced before Junos OS Release 7.4.</p> <p>pop-pop, pop-swap, push-push, swap-push, and swap-swap statements added in Junos OS Release 8.1.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.</p>
Description	<p>For Gigabit Ethernet IQ, 10-Port 10-Gigabit Ethernet SFPP interfaces, 100-Gigabit Ethernet Type 5 PIC with CFP only, Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces, define the rewrite operation to be applied to outgoing frames on this logical interface.</p> <p>The remaining statements are explained separately.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Stacking and Rewriting Gigabit Ethernet VLAN Tags</i> • input-vlan-map on page 363 • <i>Configuring Q-in-Q Tunneling (CLI Procedure)</i>

pop

Syntax	pop;
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> input-vlan-map], [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> output-vlan-map], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> input-vlan-map], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> output-vlan-map]
Release Information	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 12.3R2 for EX Series switches. Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.
Description	<div>  <p>NOTE: On EX4300 switches, pop is not supported at the [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> input-vlan-map] hierarchy level.</p> </div> <p>For Gigabit Ethernet IQ, 10-Gigabit Ethernet IQ2, and IQ2-E interfaces; 10-Gigabit Ethernet LAN/WAN PIC; aggregated Ethernet interfaces using Gigabit Ethernet IQ interfaces; 100-Gigabit Ethernet Type 5 PIC with CFP; and Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces, specify the VLAN rewrite operation to remove a VLAN tag from the top of the VLAN tag stack. The outer VLAN tag of the frame is removed.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • <i>Removing a VLAN Tag</i> • <i>Configuring Q-in-Q Tunneling (CLI Procedure)</i>



push

Syntax	<code>push;</code>
Hierarchy Level	<code>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> input-vlan-map],</code> <code>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> output-vlan-map],</code> <code>[edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i></code> <code> input-vlan-map],</code> <code>[edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i></code> <code> output-vlan-map]</code>
Release Information	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.</p>
Description	<p> NOTE: On EX4300 switches, <code>push</code> is not supported at the <code>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> output-vlan-map]</code> hierarchy level.</p> <p>Specify the VLAN rewrite operation to add a new VLAN tag to the top of the VLAN stack. An outer VLAN tag is pushed in front of the existing VLAN tag.</p> <p>You can use this statement on Gigabit Ethernet IQ and 10-Gigabit Ethernet IQ2 and IQ2-E interfaces; 10-Gigabit Ethernet LAN/WAN PIC; aggregated Ethernet interfaces using Gigabit Ethernet IQ interfaces; 100-Gigabit Ethernet Type 5 PIC with CFP; and Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces.</p> <p>If you include the push statement in the configuration, you must also include the pop statement at the <code>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> output-vlan-map]</code> hierarchy level.</p>
Required Privilege Level	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • <i>Stacking a VLAN Tag</i> • <i>Configuring Q-in-Q Tunneling (CLI Procedure)</i>

swap

Syntax	swap;
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> input-vlan-map], [edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> output-vlan-map], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> input-vlan-map], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> output-vlan-map]
Release Information	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 12.3R2 for EX Series switches. Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.
Description	<p>Specify the VLAN rewrite operation to replace a VLAN tag. The outer VLAN tag of the frame is overwritten with the user-specified VLAN tag information.</p> <p>On MX Series routers, you can enter this statement on Gigabit Ethernet IQ and 10-Gigabit Ethernet IQ2 and IQ2-E interfaces, 10-Gigabit Ethernet LAN/WAN PIC, aggregated Ethernet using Gigabit Ethernet IQ interfaces, and 100-Gigabit Ethernet Type 5 PIC with CFP. On EX Series switches, you can enter this statement on Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces.</p>
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Rewriting the VLAN Tag on Tagged Frames</i>• <i>Configuring Q-in-Q Tunneling (CLI Procedure)</i>

vlan-id-list

Syntax	<code>vlan-id-list [<i>vlan-id-numbers</i>];</code>
Hierarchy Level	<p>[edit bridge-domains <i>bridge-domain-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]</p> <p>[edit interfaces <i>interface-name</i> unit 0],</p> <p>[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i>],</p> <p>[edit vlans <i>vlan-name</i>]</p>
Release Information	<p>Statement introduced in Junos OS Release 9.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p> <p>Statement introduced in Junos OS Release 13.2 for the QFX Series.</p>
Description	<p>Specify a VLAN identifier list to use for a bridge domain or VLAN in trunk mode.</p> <p>Specify the trunk option in the interface-mode statement to accept packets with a VLAN ID that matches the list of VLAN IDs specified in the vlan-id-list statement to forward the packet within the bridge domain or VLAN configured with the matching VLAN ID. Specify the access option to accept packets with no VLAN ID to forward the packet within the bridge domain or VLAN configured with the VLAN ID that matches the VLAN ID specified in the vlan-id statement.</p> <p>This statement also enables you to bind a logical interface to a list of VLAN IDs, thereby configuring the logical interface to receive and forward a frame with a tag that matches the specified VLAN ID list.</p>
<div>  <p>WARNING: On some EX and QFX Series switches, you can apply no more than eight VLAN identifier lists to a physical interface.</p> </div>	
Options	<p><i>vlan-id-numbers</i>—Valid VLAN identifiers. You can combine individual numbers with range lists including a hyphen.</p> <p>Range: 0 through 4095</p>
<div>  <p>NOTE: On EX Series switches and the QFX Series, the range is 0 through 4094.</p> </div>	
Required Privilege Level	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>

**Related
Documentation**

- *Configuring a Bridge Domain*
- *Configuring a VLAN*
- *Configuring VLANs for EX Series Switches (CLI Procedure)*
- *Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances*
- *Configuring VLAN Identifiers for VLANs and VPLS Routing Instances*
- *Configuring Q-in-Q Tunneling (CLI Procedure)*

Reflective Relay Configuration Statement

- [reflective-relay on page 371](#)

reflective-relay

Syntax	reflective-relay;
Hierarchy Level	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> family ethernet-switching]
Release Information	Statement introduced in Junos OS Release 12.1 for the QFX Series. Statement introduced in Junos OS Release 14.1X53-D35 for EX-series.
Description	Configure a switch interface to return packets back to a device on the same interface that was used to deliver the packets.
Default	Switch interfaces are not configured for reflective relay.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• <i>Example: Configuring Reflective Relay for Use with VEPA Technology</i>• <i>Configuring Reflective Relay</i>

CHAPTER 17

Unified Forwarding Table Configuration Statements

- [forwarding-options \(chassis\) on page 374](#)
- [num-65-127-prefix on page 376](#)
- [prefix-65-127-disable on page 376](#)

forwarding-options (chassis)

Syntax forwarding options *profile-name* {
 num-65-127-prefix *value*
 lpm-profile prefix-65-127-disable
 }

Hierarchy Level [edit chassis]

Release Information Statement introduced in Junos OS Release 13.2.

Description Configure a unified forwarding table profile to allocate the amount a memory available for the following:

- MAC addresses.
- Layer 3 host entries.
- Longest prefix match table entries.

This feature enables you to select a profile that optimizes the amount of memory available for various types of forwarding-table entries based on the needs of your network. For example, for a switch that handles a great deal of Layer 2 traffic, such as a virtualized network with many servers and virtualized machines, you would choose the **l2-profile-one**, which allocates the highest amount of memory to MAC addresses.

You configure the memory allocation for LPM table entries differently, depending on whether you using Junos OS Release 13.2X51-D10 or Junos OS Release 13.2X51-D15 and later. For more information about configuring memory allocation for LPM table entries, see [“Configuring the Unified Forwarding Table” on page 248](#).

The **num-65-127-prefix *number*** statement is not supported on the **lpm-profile**. The **prefix-65-127-disable** statement is supported only on the **lpm-profile**.

When you commit a configuration with a forwarding table profile change, the Packet Forwarding Engine automatically restarts to apply the new parameters, which brings the data interfaces down and then up again.

However, starting with Junos OS Release 14.1X53-D40, upon configuring and committing a unified forwarding table profile change in a Virtual Chassis or Virtual Chassis Fabric (VCF), the Packet Forwarding Engine in member switches does not automatically restart. This behavior avoids having Virtual Chassis or VCF instability and a prolonged convergence period if a profile change is propagated to member switches and multiple Packet Forwarding Engines all restart at the same time. Instead, when you initially commit a profile configuration change, the message **Reboot required for configuration to take effect** is displayed at the master switch CLI prompt, notifying you that the profile change does not take effect until the next time you restart the Virtual Chassis or VCF. The profile configuration change is propagated to member switches that support this feature, and a reminder that a reboot is required to apply this pending configuration change appears in the system log of the master switch and applicable member switches. You then enable the profile change during a planned downtime period using the **request system reboot**

command, which quickly establishes a stable Virtual Chassis or VCF with the new configuration.



NOTE: You should plan to make unified forwarding table profile changes only when you are ready to perform a Virtual Chassis or VCF system reboot *immediately* after committing the configuration update. Otherwise, in the intervening period between committing the configuration change and rebooting the Virtual Chassis or VCF, the system can become inconsistent if a member experiences a problem and restarts. In that case, the new configuration takes effect on the member that was restarted, while the change is not yet activated on all the other members.

Options **profile-name**—name of the profile to use for memory allocation in the unified forwarding table. [Table 33 on page 375](#) lists the profiles you can choose and the associated values for each type of entry.

Table 33: Unified Forwarding Table Profiles

Profile Name	MAC Table	Host Table (unicast and multicast addresses)					
	MAC Addresses	IPv4 unicast	IPv6 unicast	IPv4 (*, G)	IPv4 (S, G)	IPv6 (*, G)	IPv6 (S, G)
l2-profile-one	288K	16K	8K	8K	8K	4K	4K
l2-profile-two	224K	80K	40K	40K	40K	20K	20K
l2-profile-three (default)	160K	144K	72K	72K	72K	36K	36K
l3-profile	96K	208K	104K	104K	104K	52K	52K
lpm-profile*	32K	16K	8K	8K	8K	4K	4K

* This profile supports only IPv4 in Junos OS 13.2X51-D10. Starting in Junos OS Release 13.2X51-D15, the **lpm-profile** supports IPv4 and IPv6 entries.

Note that if the host stores the maximum number of entries for any given type, the entire table is full and is unable to accommodate *any* entries of any other type. For information about valid combinations of table entries see [“Understanding the Unified Forwarding Table” on page 245](#).

You configure the memory allocation for LPM table entries differently depending on whether you use Junos OS 13.2X51-D10 or Junos OS 13.2X51-D15 and later. To learn how to configure memory allocation for LPM table entries see [“Configuring the Unified Forwarding Table” on page 248](#).

**Required Privilege
Level**

- Related Documentation**
- [Understanding the Unified Forwarding Table on page 245](#)
 - [Configuring the Unified Forwarding Table on page 248](#)

num-65-127-prefix

Syntax	num-65-127-prefix <i>value</i>
Hierarchy Level	[edit chassis forwarding-options <i>profile-name</i>]
Release Information	Statement introduced in Junos 13.2 for the QFX Series.
Description	Configure the number of supported IPv6 prefixes in the range /65 through /127.
Options	<p>value—With Junos OS 13.2X51D10: Value in the range 1 through 128. Each increment adds support for 16 IPv6 addresses with prefixes between /65 and /127, for a maximum of 2048 such addresses (16 x 128 = 2048).</p> <p>value—With Junos OS 13.2X51D15: Value in the range 0 through 4. Each increment adds support for 1K IPv6 addresses with prefixes between /65 and /127, for a maximum of 4K such addresses.</p>
Required Privilege Level	
Related Documentation	<ul style="list-style-type: none">• Configuring the Unified Forwarding Table on page 248

prefix-65-127-disable


Syntax	prefix-65-127-disable
Hierarchy Level	[edit chassis forwarding-options lpm-profile]
Release Information	Statement introduced in Junos 13.2X51-D15 for the QFX Series.
Description	Disable support in the longest prefix match (LPM) table for IPv6 prefixes in the range /65 through /127.
Required Privilege Level	
Related Documentation	<ul style="list-style-type: none">• Configuring the Unified Forwarding Table on page 248

CHAPTER 18

Bridging and VLANs Monitoring Commands

- `clear ethernet-switching table`
- `show ethernet-switching interface`
- `show ethernet-switching interfaces`
- `show ethernet-switching table`
- `show system statistics arp`
- `show vlans`

clear ethernet-switching table

Syntax	clear ethernet-switching table <interface <i>interface-name</i> > <mac <i>mac-address</i> > <management-vlan> <persistent-mac < <i>interface</i> <i>mac-address</i> >> <vlan <i>vlan-name</i> >
Syntax (QFX Series)	clear ethernet-switching table <interface <i>interface-name</i> > <mac <i>mac-address</i> > <persistent-mac < <i>interface</i> <i>mac-address</i> >> <vlan <i>vlan-name</i> >
Release Information	Command introduced in Junos OS Release 9.3 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	<div>  <p>NOTE: On a QFabric system, using this command on an FCoE-enabled VLAN when FCoE sessions are active can cause traffic flooding and FCoE traffic drop. The FCoE sessions are not terminated and the traffic reconverges after a short period of time.</p> </div> <p>Clear learned entries, which are media access control (MAC) addresses, in the Ethernet switching table (also called the forwarding database table).</p>
Options	<p>none—Clear learned entries in the Ethernet switching table, except for persistent MAC addresses.</p> <p>interface <i>interface-name</i>—(Optional) Clear all learned MAC addresses for the specified interface from the Ethernet switching table.</p> <p>mac <i>mac-address</i>—(Optional) Clear the specified learned MAC address from the Ethernet switching table.</p> <p>management-vlan—(Optional) Clear all MAC addresses learned for the management VLAN from the Ethernet switching table. Note that you do not specify a VLAN name because only one management VLAN exists.</p> <p>persistent-mac <<i>interface</i> <i>mac-address</i>>—(Optional) Clear all MAC addresses, including persistent MAC addresses. Use the interface option to clear all MAC addresses on an interface, or use the mac-address option to clear all entries for a specific MAC address.</p> <p>Use this command whenever you move a device in your network that has a persistent MAC address on the switch. If you move the device to another port on the switch and do not clear the persistent MAC address from the original port it was learned on, then the new port will not learn the MAC address and the device will not be able to connect. If the original port is down when you move the device, then the new port</p>

will learn the MAC address and the device can connect—however, unless you cleared the MAC address on the original port, when the port comes back up, the system reinstalls the persistent MAC address in the forwarding table for that port. If this occurs, the address is removed from the new port and the device loses connectivity.

vlan *vlan-name*—(Optional) Clear all MAC addresses learned for the specified VLAN from the Ethernet switching table.

Required Privilege Level

view

Related Documentation

- *show ethernet-switching table*
- [show ethernet-switching table on page 387](#)
- *Verifying That Persistent MAC Learning Is Working Correctly*

List of Sample Output [clear ethernet-switching table on page 379](#)

Output Fields This command produces no output.

Sample Output

[clear ethernet-switching table](#)

```
user@switch> clear ethernet-switching table
```

show ethernet-switching interface

Syntax	show ethernet-switching interface <brief detail extensive> <interface-name>
Release Information	Command introduced in Junos OS Release 12.3R2. Command introduced in Junos OS Release 12.3R2 for EX Series switches. Command introduced in Junos OS Release 13.2x51 for QFX Series switches.
Description	Display Layer 2 learning information for all the interfaces.
Options	none —Display Ethernet-switching information for all interfaces. brief detail extensive —(Optional) Display the specified level of output. interface-name —(Optional) Display Ethernet-switching information for the specified interface.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • autostate-exclude on page 273
List of Sample Output	show ethernet switching interface (Specific Interface) on page 381 show ethernet-switching interface detail on page 382 show ethernet-switching interface xe-0/0/2.0 (autostate-exclude enabled on QFX5100 switch) on page 382
Output Fields	Table 34 on page 380 describes the output fields for the show ethernet-switching interface command. Output fields are listed in the approximate order in which they appear.

Table 34: show ethernet-switching interface Output Fields

Field Name	Field Description
Logical interface	Name of the logical interface.
VLAN members	VLANs associated with this interface.
Tag	VLAN ID.
MAC limit	Number of MAC addresses that can be associated with the interface.
STP state	Spanning Tree protocol (STP) state.

Table 34: show ethernet-switching interface Output Fields (*continued*)

Field Name	Field Description
Logical interface flags	Status of Layer 2 learning properties for each interface: <ul style="list-style-type: none"> • DL—MAC learning is disabled. • LH—MAC interface limit has been reached. • AD—Packets are dropped after the MAC interface limit is reached. • DN—The MAC interface is down. • MMAS—The MAC interface is disabled after a MAC address move. • SCTL—The MAC interface is disabled after a configured storm-control level is exceeded. • AS—This interface is not included in the state calculation for VLAN members.
Tagging	Tagging state of the VLAN.

Sample Output

show ethernet switching interface (Specific Interface)

```

user@host> show ethernet-switching interface ae10.0
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down)

```

Logical interface	Vlan members	TAG	MAC limit	STP state	Logical interface flags	Tagging
ae10.0			8192			tagged
	VLAN70..	701	1024	Forwarding		
	VLAN70..	702	1024	Forwarding		
	VLAN70..	703	1024	Forwarding		
	VLAN70..	704	1024	Forwarding		
	VLAN70..	705	1024	Forwarding		
	VLAN70..	706	1024	Forwarding		
	VLAN70..	707	1024	Forwarding		
	VLAN70..	708	1024	Forwarding		
	VLAN70..	709	1024	Forwarding		
	VLAN71..	710	1024	Forwarding		
	VLAN71..	711	1024	Forwarding		
	VLAN71..	712	1024	Forwarding		
	VLAN71..	713	1024	Forwarding		
	VLAN71..	714	1024	Forwarding		

```
VLAN71.. 715
[...output truncated...]
```

show ethernet-switching interface detail

```
user@host> show ethernet-switching interface detail
Information for interface family:
Name: ge-1/0/3.0
  Type: IFF                                Handle: 0x8bba280
  Index: 331                              Generation: 159
                                           Flags: UP,
                                           Routing/Vlan index: 4
                                           Address family: 50
                                           MAC sequence number: 0
                                           MACs learned: 0
                                           Non configured static MACs learned: 0
  IFD index: 141
  IFL index: 331
  Sequence number: 0
  MAC limit: 65535
  Static MACs learned: 0
Name: ge-1/0/3.0
  Type: IFBD (static)                     Handle: 0x8bb6e00
  Index:                                  Generation: 129
                                           Flags: UP,
                                           Routing/Vlan index: 2
                                           Address family:
                                           MAC sequence number: 1
                                           MACs learned: 0
                                           Non configured static MACs learned: 0
                                           Rewrite op:
  Trunk id: 0
  IFD index:
  IFL index:
  Sequence number: 1
  MAC limit: 65535
  Static MACs learned: 0
  VSTP index: 11
Name: ge-1/0/3.0
  Type: IFBD (static)                     Handle: 0x8bb6f00
  Index:                                  Generation: 130
                                           Flags: UP,
                                           Routing/Vlan index: 3
                                           Address family:
                                           MAC sequence number: 1
                                           MACs learned: 0
                                           Non configured static MACs learned: 0
                                           Rewrite op:
  Trunk id: 0
  IFD index:
  IFL index:
  Sequence number: 1
  MAC limit: 65535
  Static MACs learned: 0
  VSTP index: 11
```

show ethernet-switching interface xe-0/0/2.0 (autostate-exclude enabled on QFX5100 switch)

```
user@switch> show ethernet-switching interface xe-0/0/2.0

Routing Instance Name : default-switch
Logical Interface flags (DL - disable learning, AD - packet action drop,
                        LH - MAC limit hit, DN - interface down,
                        MMAS - Mac-move action shutdown, AS - Autostate-exclude enabled
                        SCTL - shutdown by Storm-control)
```

Logical interface	Vlan members	TAG	MAC limit	STP state	Logical interface flags	Tagging
xe-0/0/2.0	v100	100	294912	Forwarding	AS	tagged

show ethernet-switching interfaces

Syntax	show ethernet-switching interfaces <brief detail summary> <interface <i>interface-name</i> >
Release Information	Command introduced in Junos OS Release 11.1 for the QFX Series. Command introduced in Junos OS Release 14.1X53-D20 for OCX Series switches.
Description	Display information about switched Ethernet interfaces.
Options	<p>none—(Optional) Display brief information for Ethernet-switching interfaces.</p> <p>brief detail summary—(Optional) Display the specified level of output.</p> <p>interface <i>interface-name</i>—(Optional) Display Ethernet-switching information for a specific interface.</p>
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • Troubleshooting Ethernet Switching on page 84Understanding Bridging and VLANs on page 35 • Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43 • Example: Setting Up Bridging with Multiple VLANs • Understanding FCoE • Interfaces Overview
List of Sample Output	show ethernet-switching interfaces on page 384 show ethernet-switching interfaces summary on page 385 show ethernet-switching interfaces brief on page 385 show ethernet-switching interfaces detail on page 385 show ethernet-switching interfaces interface-name on page 386
Output Fields	Table 35 on page 383 lists the output fields for the show ethernet-switching interfaces command. Output fields are listed in the approximate order in which they appear.

Table 35: show ethernet-switching interfaces Output Fields

Field Name	Field Description	Level of Output
Interface	Name of a switching interface.	All levels
State	Interface state. Values are up or down .	none, brief , detail , summary
VLAN members	Name of a VLAN.	none, brief , detail , summary

Table 35: show ethernet-switching interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
Blocking	Forwarding state of the interface: <ul style="list-style-type: none"> • blocked—Traffic is not being forwarded on the interface. • unblocked—Traffic is forwarded on the interface. • MAC limit exceeded—The interface is temporarily disabled because of a MAC limiting error. The disabled interface is automatically restored to service when the disable timeout expires. • MAC move limit exceeded—The interface is temporarily disabled because of a MAC move limiting error. The disabled interface is automatically restored to service when the disable timeout expires. • Storm control in effect —The interface is temporarily disabled because of a storm control error. The disabled interface is automatically restored to service when the disable timeout expires. • Storm control shutdown in effect —The interface is temporarily disabled because of a storm control shutdown error. The disabled interface is automatically restored to service when the disable timeout expires. 	none, brief , detail , summary
Index	VLAN index internal to Junos OS software.	detail
untagged tagged	Specifies whether the interface forwards IEEE802.1Q-tagged or untagged traffic.	detail

Sample Output

show ethernet-switching interfaces

```
user@switch> show ethernet-switching interfaces
```

```

Interface  State  VLAN members  Blocking
xe-0/0/0.0  up    T1122         unblocked
xe-0/0/1.0  down  default       - MAC limit exceeded
xe-0/0/2.0  down  default       - MAC move limit exceeded
xe-0/0/3.0  down  default       - Storm control in effect
xe-0/0/4.0  down  default       unblocked
xe-0/0/5.0  down  default       unblocked
xe-0/0/6.0  down  default       unblocked
xe-0/0/7.0  down  default       unblocked
xe-0/0/8.0  down  default       unblocked
xe-0/0/9.0  up    T111         unblocked
xe-0/0/10.0 down  default       unblocked
xe-0/0/11.0 down  default       unblocked
xe-0/0/12.0 down  default       unblocked
xe-0/0/13.0 down  default       unblocked
xe-0/0/14.0 down  default       unblocked
xe-0/0/15.0 down  default       unblocked
xe-0/0/16.0 down  default       unblocked
xe-0/0/17.0 down  default       unblocked
xe-0/0/18.0 down  default       unblocked
xe-0/0/19.0 up    T111         unblocked
xe-0/1/0.0  down  default       unblocked
xe-0/1/1.0  down  default       unblocked
xe-0/1/2.0  down  default       unblocked
xe-0/1/3.0  down  default       unblocked

```

show ethernet-switching interfaces summary

```

user@switch> show ethernet-switching interfaces summary
xe-0/0/0.0
xe-0/0/1.0
xe-0/0/2.0
xe-0/0/3.0
xe-0/0/8.0
xe-0/0/10.0
xe-0/0/11.0

```

show ethernet-switching interfaces brief

```

user@switch> show ethernet-switching interfaces brief
Interface  State  VLAN members  Blocking
xe-0/0/0.0  down  default        unblocked
xe-0/0/1.0  down  employee-vlan  unblocked
xe-0/0/2.0  down  employee-vlan  unblocked
xe-0/0/3.0  down  employee-vlan  unblocked
xe-0/0/8.0  down  employee-vlan  unblocked
xe-0/0/10.0 down  default        unblocked
xe-0/0/11.0 down  employee-vlan  unblocked

```

show ethernet-switching interfaces detail

```

user@switch> show ethernet-switching interfaces detail
Interface: xe-0/0/0.0 Index: 65
State: down
VLANs:
    default                untagged    unblocked

Interface: xe-0/0/1.0 Index: 66
State: down
VLANs:
    employee-vlan          untagged    unblocked

Interface: xe-0/0/2.0 Index: 67
State: down
VLANs:
    employee-vlan          untagged    unblocked

Interface: xe-0/0/3.0 Index: 68
State: down
VLANs:
    employee-vlan          untagged    unblocked

Interface: xe-0/0/8.0 Index: 69
State: down
VLANs:
    employee-vlan          untagged    unblocked

Interface: xe-0/0/10.0 Index: 70
State: down
VLANs:
    default                untagged    unblocked

Interface: xe-0/0/11.0 Index: 71
State: down
VLANs:
    employee-vlan          tagged      unblocked

```

show ethernet-switching interfaces interface-name

```
user@switch> show ethernet-switching interfaces xe-0/0/0.0
Interface  State  VLAN members  Blocking
xe-0/0/0.0 down   default        unblocked
```


show ethernet-switching table

Syntax	<pre>show ethernet-switching table <brief detail extensive summary> <interface <i>interface-name</i>> <management-vlan> <sort-by (<i>name</i> <i>tag</i>)> <vlan <i>vlan-name</i>></pre>	
Release Information	<p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>Output for private VLANs introduced in Junos OS Release 12.1 for the QFX Series.</p>	
Description	Displays the Ethernet switching table.	
Options	<p>none—(Optional) Display brief information about the Ethernet switching table.</p> <p>brief detail extensive summary—(Optional) Display the specified level of output.</p> <p>interface <i>interface-name</i>—(Optional) Display the Ethernet switching table for a specific interface.</p> <p>management-vlan—(Optional) Display the Ethernet switching table for a management VLAN.</p> <p>sort-by (<i>name</i> <i>tag</i>)—(Optional) Display VLANs in ascending order of VLAN IDs or VLAN names.</p> <p>vlan <i>vlan-name</i>—(Optional) Display the Ethernet switching table for a specific VLAN.</p>	
Required Privilege Level	view	
Related Documentation	<ul style="list-style-type: none"> • Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43 • Example: Setting Up Bridging with Multiple VLANs 	
List of Sample Output	<p>show ethernet-switching table (Enhanced Layer 2 Software) on page 388</p> <p>show ethernet-switching table on page 389</p> <p>show ethernet-switching table (Private VLANs) on page 390</p> <p>show ethernet-switching table brief on page 390</p> <p>show ethernet-switching table detail on page 391</p> <p>show ethernet-switching table extensive on page 392</p> <p>show ethernet-switching table interface on page 394</p>	
Output Fields	<p>Table 36 on page 387 lists the output fields for the show ethernet-switching table command. Output fields are listed in the approximate order in which they appear.</p>	

Table 36: show ethernet-switching table Output Fields

Field Name	Field Description	Level of Output
VLAN	Name of a VLAN.	All levels

Table 36: show ethernet-switching table Output Fields (*continued*)

Field Name	Field Description	Level of Output
MAC address	MAC address associated with the VLAN.	All levels
Type	Type of MAC address: <ul style="list-style-type: none"> • static—The MAC address is manually created. • learn—The MAC address is learned dynamically from a packet's source MAC address. • flood—The MAC address is unknown and flooded to all members. 	All levels
Age	Time remaining before the entry ages out and is removed from the Ethernet switching table.	All levels
Interfaces	Interface associated with learned MAC addresses or with the All-members option (flood entry).	All levels
Learned	For learned entries, the time at which the entry was added to the Ethernet switching table.	detail, extensive

Sample Output

show ethernet-switching table (Enhanced Layer 2 Software)

```
user@switch> show ethernet-switching table
MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent
static
          SE - statistics enabled, NM - non configured MAC, R - remote PE MAC,
0 - ovsdb MAC)
```

```
Ethernet switching table : 2 entries, 2 learned
Routing instance : default-switch
  Vlan      MAC      MAC      Age   Logical
  name      address  flags
vlan1      b0:c6:9a:ca:3c:01  D        -    ae1.0

  vlan1      b0:c6:9a:ca:3c:03  D        -    ae1.0
```

```
MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent
static
          SE - statistics enabled, NM - non configured MAC, R - remote PE MAC,
0 - ovsdb MAC)
```

```
Ethernet switching table : 2 entries, 2 learned
Routing instance : default-switch
  Vlan      MAC      MAC      Age   Logical
  name      address  flags
vlan10     b0:c6:9a:ca:3c:01  D        -    ae1.0

  vlan10     b0:c6:9a:ca:3c:03  D        -    ae1.0
```

MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent static

SE - statistics enabled, NM - non configured MAC, R - remote PE MAC, O - ovsdb MAC)

Ethernet switching table : 2 entries, 2 learned

Routing instance : default-switch

Vlan name	MAC address	MAC flags	Age	Logical interface
vlan2	b0:c6:9a:ca:3c:01	D	-	ae1.0
vlan2	b0:c6:9a:ca:3c:03	D	-	ae1.0

MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent static

SE - statistics enabled, NM - non configured MAC, R - remote PE MAC, O - ovsdb MAC)

Ethernet switching table : 2 entries, 2 learned

Routing instance : default-switch

Vlan name	MAC address	MAC flags	Age	Logical interface
vlan3	b0:c6:9a:ca:3c:01	D	-	ae1.0
vlan3	b0:c6:9a:ca:3c:03	D	-	ae1.0

MAC flags (S - static MAC, D - dynamic MAC, L - locally learned, P - Persistent static

SE - statistics enabled, NM - non configured MAC, R - remote PE MAC, O - ovsdb MAC)

Ethernet switching table : 2 entries, 2 learned

Routing instance : default-switch

Vlan name	MAC address	MAC flags	Age	Logical interface
vlan4	b0:c6:9a:ca:3c:01	D	-	ae1.0
vlan4	b0:c6:9a:ca:3c:03	D	-	ae1.0

show ethernet-switching table

user@switch> show ethernet-switching table

Ethernet-switching table: 57 entries, 17 learned

VLAN	MAC address	Type	Age	Interfaces
F2	*	Flood	-	All-members
F2	00:00:05:00:00:03	Learn	0	xe-0/0/44.0
F2	00:19:e2:50:7d:e0	Static	-	Router
Linux	*	Flood	-	All-members
Linux	00:19:e2:50:7d:e0	Static	-	Router
Linux	00:30:48:90:54:89	Learn	0	xe-0/0/47.0
T1	*	Flood	-	All-members
T1	00:00:05:00:00:01	Learn	0	xe-0/0/46.0
T1	00:00:5e:00:01:00	Static	-	Router
T1	00:19:e2:50:63:e0	Learn	0	xe-0/0/46.0
T1	00:19:e2:50:7d:e0	Static	-	Router
T10	*	Flood	-	All-members

```

T10      00:00:5e:00:01:09 Static      - Router
T10      00:19:e2:50:63:e0 Learn       0 xe-0/0/46.0
T10      00:19:e2:50:7d:e0 Static      - Router
T111     *                               Flood      - All-members
T111     00:19:e2:50:63:e0 Learn       0 xe-0/0/15.0
T111     00:19:e2:50:7d:e0 Static      - Router
T111     00:19:e2:50:ac:00 Learn       0 xe-0/0/15.0
T2       *                               Flood      - All-members
T2       00:00:5e:00:01:01 Static      - Router
T2       00:19:e2:50:63:e0 Learn       0 xe-0/0/46.0
T2       00:19:e2:50:7d:e0 Static      - Router
T3       *                               Flood      - All-members
T3       00:00:5e:00:01:02 Static      - Router
T3       00:19:e2:50:63:e0 Learn       0 xe-0/0/46.0
T3       00:19:e2:50:7d:e0 Static      - Router
T4       *                               Flood      - All-members
T4       00:00:5e:00:01:03 Static      - Router
T4       00:19:e2:50:63:e0 Learn       0 xe-0/0/46.0

```

[output truncated]

show ethernet-switching table (Private VLANs)

```

user@switch> show ethernet-switching table
Ethernet-switching table: 10 entries, 3 learned

```

VLAN	MAC address	Type	Age	Interfaces
pvlan	*	Flood		- All-members
pvlan	00:10:94:00:00:02	Replicated		- xe-0/0/28.0
pvlan	00:10:94:00:00:35	Replicated		- xe-0/0/46.0
pvlan	00:10:94:00:00:46	Replicated		- xe-0/0/4.0
c2	*	Flood		- All-members
c2	00:10:94:00:00:02	Learn	0	xe-0/0/28.0
c1	*	Flood		- All-members
c1	00:10:94:00:00:46	Learn	0	xe-0/0/4.0
__pvlan_pvlan_xe-0/0/46.0__	*	Flood		- All-members
__pvlan_pvlan_xe-0/0/46.0__	00:10:94:00:00:35	Learn	0	xe-0/0/46.0

show ethernet-switching table brief

```

user@switch> show ethernet-switching table brief
Ethernet-switching table: 57 entries, 17 learned

```

VLAN	MAC address	Type	Age	Interfaces
F2	*	Flood		- All-members
F2	00:00:05:00:00:03	Learn	0	xe-0/0/44.0
F2	00:19:e2:50:7d:e0	Static		- Router
Linux	*	Flood		- All-members
Linux	00:19:e2:50:7d:e0	Static		- Router
Linux	00:30:48:90:54:89	Learn	0	xe-0/0/47.0
T1	*	Flood		- All-members
T1	00:00:05:00:00:01	Learn	0	xe-0/0/46.0
T1	00:00:5e:00:01:00	Static		- Router
T1	00:19:e2:50:63:e0	Learn	0	xe-0/0/46.0
T1	00:19:e2:50:7d:e0	Static		- Router
T10	*	Flood		- All-members
T10	00:00:5e:00:01:09	Static		- Router
T10	00:19:e2:50:63:e0	Learn	0	xe-0/0/46.0
T10	00:19:e2:50:7d:e0	Static		- Router
T111	*	Flood		- All-members
T111	00:19:e2:50:63:e0	Learn	0	xe-0/0/15.0
T111	00:19:e2:50:7d:e0	Static		- Router
T111	00:19:e2:50:ac:00	Learn	0	xe-0/0/15.0
T2	*	Flood		- All-members

```

T2          00:00:5e:00:01:01 Static      - Router
T2          00:19:e2:50:63:e0 Learn       0 xe-0/0/46.0
T2          00:19:e2:50:7d:e0 Static      - Router
T3          *                               Flood    - All-members
T3          00:00:5e:00:01:02 Static      - Router
T3          00:19:e2:50:63:e0 Learn       0 xe-0/0/46.0
T3          00:19:e2:50:7d:e0 Static      - Router
T4          *                               Flood    - All-members
T4          00:00:5e:00:01:03 Static      - Router
T4          00:19:e2:50:63:e0 Learn       0 xe-0/0/46.0
[output truncated]

```

show ethernet-switching table detail

```

user@switch> show ethernet-switching table detail
Ethernet-switching table: 57 entries, 17 learned
F2, *
  Interface(s): xe-0/0/44.0
  Type: Flood
  Nexthop index: 0

F2, 00:00:05:00:00:03
  Interface(s): xe-0/0/44.0
  Type: Learn, Age: 0, Learned: 2:03:09
  Nexthop index: 0

F2, 00:19:e2:50:7d:e0
  Interface(s): Router
  Type: Static
  Nexthop index: 0

Linux, *
  Interface(s): xe-0/0/47.0
  Type: Flood
  Nexthop index: 0

Linux, 00:19:e2:50:7d:e0
  Interface(s): Router
  Type: Static
  Nexthop index: 0

Linux, 00:30:48:90:54:89
  Interface(s): xe-0/0/47.0
  Type: Learn, Age: 0, Learned: 2:03:08
  Nexthop index: 0

T1, *
  Interface(s): xe-0/0/46.0
  Type: Flood
  Nexthop index: 0

T1, 00:00:05:00:00:01
  Interface(s): xe-0/0/46.0
  Type: Learn, Age: 0, Learned: 2:03:07
  Nexthop index: 0

T1, 00:00:5e:00:01:00
  Interface(s): Router
  Type: Static
  Nexthop index: 0

```

```
T1, 00:19:e2:50:63:e0
  Interface(s): xe-0/0/46.0
  Type: Learn, Age: 0, Learned: 2:03:07
  Nexthop index: 0

T1, 00:19:e2:50:7d:e0
  Interface(s): Router
  Type: Static
  Nexthop index: 0

T10, *
  Interface(s): xe-0/0/46.0
  Type: Flood
  Nexthop index: 0

T10, 00:00:5e:00:01:09
  Interface(s): Router
  Type: Static
  Nexthop index: 0

T10, 00:19:e2:50:63:e0
  Interface(s): xe-0/0/46.0
  Type: Learn, Age: 0, Learned: 2:03:08
  Nexthop index: 0

T10, 00:19:e2:50:7d:e0
  Interface(s): Router
  Type: Static
  Nexthop index: 0

T111, *
  Interface(s): xe-0/0/15.0
  Type: Flood
  Nexthop index: 0
[output truncated]
```

show ethernet-switching table extensive

```
user@switch> show ethernet-switching table extensive
Ethernet-switching table: 57 entries, 17 learned
F2, *
  Interface(s): xe-0/0/44.0
  Type: Flood
  Nexthop index: 0

F2, 00:00:05:00:00:03
  Interface(s): xe-0/0/44.0
  Type: Learn, Age: 0, Learned: 2:03:09
  Nexthop index: 0

F2, 00:19:e2:50:7d:e0
  Interface(s): Router
  Type: Static
  Nexthop index: 0

Linux, *
  Interface(s): xe-0/0/47.0
  Type: Flood
  Nexthop index: 0

Linux, 00:19:e2:50:7d:e0
```

```
Interface(s): Router
Type: Static
Nexthop index: 0

Linux, 00:30:48:90:54:89
Interface(s): xe-0/0/47.0
Type: Learn, Age: 0, Learned: 2:03:08
Nexthop index: 0

T1, *
Interface(s): xe-0/0/46.0
Type: Flood
Nexthop index: 0

T1, 00:00:05:00:00:01
Interface(s): xe-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:07
Nexthop index: 0

T1, 00:00:5e:00:01:00
Interface(s): Router
Type: Static
Nexthop index: 0

T1, 00:19:e2:50:63:e0
Interface(s): xe-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:07
Nexthop index: 0

T1, 00:19:e2:50:7d:e0
Interface(s): Router
Type: Static
Nexthop index: 0

T10, *
Interface(s): xe-0/0/46.0
Type: Flood
Nexthop index: 0

T10, 00:00:5e:00:01:09
Interface(s): Router
Type: Static
Nexthop index: 0

T10, 00:19:e2:50:63:e0
Interface(s): xe-0/0/46.0
Type: Learn, Age: 0, Learned: 2:03:08
Nexthop index: 0

T10, 00:19:e2:50:7d:e0
Interface(s): Router
Type: Static
Nexthop index: 0

T111, *
Interface(s): xe-0/0/15.0
Type: Flood
Nexthop index: 0
[output truncated]
```

show ethernet-switching table interface

```
user@switch> show ethernet-switching table interface xe-0/0/1
Ethernet-switching table: 1 unicast entries
VLAN      MAC address      Type      Age Interfaces
V1        *                Flood     - All-members
V1        00:00:05:00:00:05 Learn       0 xe-0/0/1.0
```


show system statistics arp

Syntax	show system statistics arp
Release Information	Command introduced in Junos OS Release 9.6 for EX Series switches.
Description	Display system-wide Address Resolution Protocol (ARP) statistics.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • <i>Example: Configuring Proxy ARP on an EX Series Switch</i> • Verifying That Proxy ARP Is Working Correctly on page 231

show system statistics arp

```

user@switch> show system statistics arp
arp:
    90060 datagrams received
    34 ARP requests received
    610 ARP replies received
    0 resolution request received
    0 unrestricted proxy requests
    0 restricted proxy requests
    0 received proxy requests
    0 unrestricted proxy requests not proxied
    0 restricted proxy requests not proxied
    0 datagrams with bogus interface
    0 datagrams with incorrect length
    0 datagrams for non-IP protocol
    0 datagrams with unsupported op code
    0 datagrams with bad protocol address length
    0 datagrams with bad hardware address length
    0 datagrams with multicast source address
    0 datagrams with multicast target address
    0 datagrams with my own hardware address
    0 datagrams for an address not on the interface
    0 datagrams with a broadcast source address
    294 datagrams with source address duplicate to mine
    89113 datagrams which were not for me
    0 packets discarded waiting for resolution
    0 packets sent after waiting for resolution
    309 ARP requests sent
    35 ARP replies sent
    0 requests for memory denied
    0 requests dropped on entry
    0 requests dropped during retry
    0 requests dropped due to interface deletion
    0 requests on unnumbered interfaces
    0 new requests on unnumbered interfaces
    0 replies for from unnumbered interfaces
    0 requests on unnumbered interface with non-subnetted donor
    0 replies from unnumbered interface with non-subnetted donor

```

show vlans

Syntax `show vlans`
`<brief | detail | extensive>`
`<dot1q-tunneling>`
`<sort-by (tag | name)>`
`<vlan-range-name>`

Release Information Command introduced in Junos OS Release 11.1 for the QFX Series.
Option **dot1q-tunneling** added in Junos OS Release 12.1 for the QFX Series.

Description Display information about VLANs configured on bridged Ethernet interfaces. For interfaces configured to support a VoIP VLAN and a data VLAN, the **show vlans** command displays both tagged and untagged membership for those VLANs.



NOTE: When a series of VLANs is created using the `vlan-range` statement, such VLAN names are preceded and followed by a double underscore. For example, a series of VLANs using the VLAN range 1 through 3 and the base VLAN name `marketing` would be displayed as `__marketing_1__`, `__marketing_2__`, and `__marketing_3__`.



NOTE: To display an 802.1X supplicant successfully authenticated in multiple-supplicant mode with dynamic VLAN movement, use the `show vlans vlan-name extensive` operational mode command, where `vlan-name` is the dynamic VLAN.

Options **none**—Display information for all VLANs. VLAN information is displayed by VLAN name in ascending order.

brief | detail | extensive—(Optional) Display the specified level of output.

sort-by (tag | name)—(Optional) Display VLANs in ascending order of VLAN IDs or VLAN names.

vlan-range-name—(Optional) Display VLANs in ascending order of VLAN range names.

Required Privilege Level `view`

Related Documentation

- [Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43](#)
- [Example: Setting Up Bridging with Multiple VLANs](#)
- [Understanding Bridging and VLANs](#)
- [show ethernet-switching interfaces on page 383](#)

List of Sample Output

- [show vlans on page 399](#)
- [show vlans \(Private VLANs\) on page 399](#)
- [show vlans brief on page 400](#)
- [show vlans detail on page 400](#)
- [show vlans extensive \(Port-Based\) on page 401](#)
- [show vlans \(Q-in-Q Tunneling\) on page 402](#)
- [show vlans extensive \(Q-in-Q Tunneling\) on page 402](#)
- [show vlans extensive \(Q-in-Q Tunneling and L2TP\) on page 402](#)
- [show vlans sort-by tag on page 402](#)
- [show vlans sort-by name on page 403](#)
- [show vlans tag on page 404](#)

Output Fields Table 37 on page 397 lists the output fields for the **show vlans** command. Output fields are listed in the approximate order in which they appear.

Table 37: show vlans Output Fields

Field Name	Field Description	Level of Output
Name	Name of a VLAN.	none, brief
Tag	802.1Q tag applied to this VLAN. If none is displayed, no tag is applied.	All levels
Interfaces	Interface associated with learned MAC addresses or All-members option (flood entry). An asterisk (*) beside the interface indicates that the interface is UP .	All levels
Address	IP address.	none, brief
Ports Active /Total	Number of interfaces associated with a VLAN: Active indicates interfaces that are UP , and Total indicates interfaces that are active and inactive.	brief
VLAN	Name of a VLAN.	detail, extensive
Admin state	State of the interface. Values are: enabled —The interface is turned on, and the physical link is operational and can pass packets.	detail,extensive
MAC learning Status	Indicates if MAC learning is disabled.	detail, extensive
Description	Description for the VLAN.	detail,extensive
Primary IP	Primary IP address associated with a VLAN.	detail
Number of interfaces	Number of interfaces associated with a VLAN. Both the total number of interfaces and the number of active interfaces associated with a VLAN are displayed.	detail, extensive
STP	Spanning tree associated with a VLAN.	detail,extensive
Tagged interfaces	Tagged interfaces with which a VLAN is associated.	detail,extensive

Table 37: show vlans Output Fields (*continued*)

Field Name	Field Description	Level of Output
Untagged interfaces	Untagged interfaces with which a VLAN is associated.	detail. extensive
Dot1q Tunneling Status	Indicates if Q-in-Q tunneling is enabled.	extensive
Customer VLAN ranges	List of customer VLAN (C-VLAN) ranges associated with this service VLAN (S-VLAN).	extensive
Private VLAN Mode	The private VLAN mode for this VLAN. Values include Primary , Isolated , and Community .	extensive
Primary VLAN	Primary VLAN tag for this secondary VLAN.	extensive
Internal Index	VLAN index internal to Junos OS software.	extensive
Origin	Manner in which the VLAN was created: static or learn .	extensive
Protocol	Port-based VLAN or MAC-based VLAN. MAC-based protocol is displayed when VLAN assignment is done either statically or dynamically through 802.1X,	extensive
IP addresses	IP address associated with a VLAN.	extensive
Number of MAC entries	For MAC-based VLANs created either statically or dynamically, the MAC addresses associated with an interface.	extensive
Number of mapping rules	Number of mapping rules for Q-in-Q tunneling (Push) and VLAN translation (Swap).	
Secondary VLANs	Secondary VLANs associated with a primary VLAN.	extensive
Isolated VLANs	Isolated VLANs associated with a primary VLAN.	extensive
Community VLANs	Community VLANs associated with a primary VLAN.	extensive
VLANs summary	VLAN counts: <ul style="list-style-type: none"> • Total—Total number of VLANs on the switch. • Configured VLANs—Number of VLANs that are based on user-configured settings. • Internal VLANs—Number of VLANs created by the system with no explicit configuration or protocol—for example, the default VLAN and the VLAN created when a trunk interface is not configured with native VLAN membership. • Temporary VLANs—Number of VLANs from the previous configuration that the system retains for a limited time after restart. Temporary VLANs are converted into one of the other types of VLAN, or are removed from the system if the current configuration does not require them. 	All levels

Table 37: show vlans Output Fields (*continued*)

Field Name	Field Description	Level of Output
Dot1q VLANs summary	802.1Q VLAN counts: <ul style="list-style-type: none"> • Total—Total number of 802.1Q-tagged and untagged VLANs on the switch. • Tagged VLANs—Number of 802.1Q-tagged VLANs. • Untagged VLANs—Number of untagged 802.1Q VLANs. • Private VLAN—Counts of the following kinds of 802.1Q private VLANs (PVLANS): <ul style="list-style-type: none"> • Primary VLANs—Number of primary forwarding private VLANs. • Community VLANs—Number of community transporting and forwarding private VLANs. • Isolated VLANs—Number of isolated receiving and forwarding private VLANs. • Inter-switch-isolated VLANs—Number of inter-switch isolated receiving and forwarding private VLANs. 	All levels
Dot1q Tunneled VLANs summary	Q-in-Q-tunneled VLAN counts: <ul style="list-style-type: none"> • Total—Total number of Q-in-Q-tunneled VLANs on the switch. • Private VLAN—Counts of primary, community, and isolated Q-in-Q-tunneled private VLANs (PVLANS). 	All levels

Sample Output

show vlans

```
user@switch> show vlans
```

Name	Tag	Interfaces
default	None	xe-0/0/34.0, xe-0/0/33.0, xe-0/0/32.0, xe-0/0/31.0, xe-0/0/30.0, xe-0/0/29.0, xe-0/0/28.0, xe-0/0/27.0, xe-0/0/26.0, xe-0/0/25.0, xe-0/0/19.0, xe-0/0/18.0, xe-0/0/17.0, xe-0/0/16.0, xe-0/0/15.0, xe-0/0/14.0, xe-0/0/13.0, xe-0/0/11.0, xe-0/0/9.0, xe-0/0/8.0, xe-0/0/3.0, xe-0/0/2.0, xe-0/0/1.0
v0001	1	xe-0/0/24.0, xe-0/0/23.0, xe-0/0/22.0, xe-0/0/21.0
v0002	2	None
v0003	3	None
v0004	4	None
v0005	5	None

show vlans (Private VLANs)

```
user@switch> show vlans
```

Name	Tag	Interfaces
__pvlan_pvlan_xe-0/0/46.0__		

```

c1                xe-0/0/44.0*, xe-0/0/46.0*
c2                xe-0/0/4.0*, xe-0/0/44.0*
default           xe-0/0/28.0*, xe-0/0/44.0*
pvlan             500
                  None
                  xe-0/0/4.0*, xe-0/0/28.0*, xe-0/0/44.0*, xe-0/0/46.0*

```

show vlans brief

```
user@switch> show vlans brief
```

Name	Tag	Address	Ports Active/Total
default	None		0/23
v0001	1		0/4
v0002	2		0/0
v0003	3		0/0
v0004	4		0/0
v0005	5		0/0
v0006	6		0/0
v0007	7		0/0
v0008	8		0/0
v0009	9		0/0
v0010	10		0/2
v0011	11		0/0
v0012	12		0/0
v0013	13		0/0
v0014	14		0/0
v0015	15		0/0
v0016	16		0/0

show vlans detail

```
user@switch> show vlans detail
```

```
VLAN: default, Tag: Untagged, Admin state: Enabled
```

```
Description: None
```

```
Primary IP: None, Number of interfaces: 23 (Active = 0)
```

```
STP: None, RTG: None
```

```
Untagged interfaces: xe-0/0/34.0, xe-0/0/33.0, xe-0/0/32.0, xe-0/0/31.0,
xe-0/0/30.0, xe-0/0/29.0, xe-0/0/28.0, xe-0/0/27.0, xe-0/0/26.0,
xe-0/0/25.0, xe-0/0/19.0, xe-0/0/18.0, xe-0/0/17.0, xe-0/0/16.0,
xe-0/0/15.0, xe-0/0/14.0, xe-0/0/13.0, xe-0/0/11.0, xe-0/0/9.0, xe-0/0/8.0,
xe-0/0/3.0, xe-0/0/2.0, xe-0/0/1.0,
```

```
Tagged interfaces: None
```

```
VLAN: v0001, Tag: 802.1Q Tag 1, Admin state: Enabled
```

```
Description: None
```

```
Primary IP: None, Number of interfaces: 4 (Active = 0)
```

```
Dot1q Tunneling Status: Enabled
```

```
STP: None, RTG: None
```

```
Untagged interfaces: None
```

```
Tagged interfaces: xe-0/0/24.0, xe-0/0/23.0, xe-0/0/22.0, xe-0/0/21.0,
```

```
VLAN: v0002, Tag: 802.1Q Tag 2, Admin state: Enabled
```

```
Description: None
```

```
Primary IP: None, Number of interfaces: 0 (Active = 0)
```

```
STP: None, RTG: None
```

```
Untagged interfaces: None
```

```
Tagged interfaces: None
```

```

VLAN: v0003, Tag: 802.1Q Tag 3, Admin state: Enabled
Description: None
Primary IP: None, Number of interfaces: 0 (Active = 0)
STP: None, RTG: None
Untagged interfaces: None
Tagged interfaces: None

VLAN: vlan4000, 802.1Q Tag: Untagged, Admin State: Enabled
MAC learning Status: Disabled
Number of interfaces: 0 (Active = 0)

```

show vlans extensive (Port-Based)

```

user@switch> show vlans extensive
VLAN: default, created at Mon Feb  4 12:13:47 2008
Tag: None, Internal index: 0, Admin state: Enabled, Origin: static
Description: None
Customer VLAN ranges:
    1-4100
Protocol: Port based
IP addresses: None
STP: None, RTG: None.
Number of interfaces: Tagged 0 (Active = 0), Untagged 23 (Active = 0)
    xe-0/0/34.0 (untagged, access)
    xe-0/0/33.0 (untagged, access)
    xe-0/0/32.0 (untagged, access)
    xe-0/0/31.0 (untagged, access)
    xe-0/0/30.0 (untagged, access)
    xe-0/0/29.0 (untagged, access)
    xe-0/0/28.0 (untagged, access)
    xe-0/0/27.0 (untagged, access)
    xe-0/0/26.0 (untagged, access)
    xe-0/0/25.0 (untagged, access)
    xe-0/0/19.0 (untagged, access)
    xe-0/0/18.0 (untagged, access)
    xe-0/0/17.0 (untagged, access)
    xe-0/0/16.0 (untagged, access)
    xe-0/0/15.0 (untagged, access)
    xe-0/0/14.0 (untagged, access)
    xe-0/0/13.0 (untagged, access)
    xe-0/0/11.0 (untagged, access)
    xe-0/0/9.0 (untagged, access)
    xe-0/0/8.0 (untagged, access)
    xe-0/0/3.0 (untagged, access)
    xe-0/0/2.0 (untagged, access)
    xe-0/0/1.0 (untagged, access)

Secondary VLANs: Isolated 1, Community 1
Isolated VLANs :
    __pvlan_pvlan_xe-0/0/3.0__
Community VLANs :
    comm1

VLAN: v0001, created at Mon Feb  4 12:13:47 2008
Tag: 1, Internal index: 1, Admin state: Enabled, Origin: static
Description: None
Protocol: Port based, Layer 3 interface: None
IP addresses: None
STP: None, RTG: None.
Number of interfaces: Tagged 4 (Active = 0), Untagged 0 (Active = 0)

```

```

xe-0/0/24.0 (tagged, trunk)
xe-0/0/23.0 (tagged, trunk)
xe-0/0/22.0 (tagged, trunk)
xe-0/0/21.0 (tagged, trunk)

```

```

VLAN: v0002, created at Mon Feb  4 12:13:47 2008
Tag: 2, Internal index: 2, Admin state: Enabled, Origin: static
Description: None
Protocol: Port based, Layer 3 interface: None
IP addresses: None
STP: None, RTG: None.
Number of interfaces: Tagged 0 (Active = 0), Untagged 0 (Active = 0)
None

```

```

VLAN: v0003, created at Mon Feb  4 12:13:47 2008
Tag: 3, Internal index: 3, Admin state: Enabled, Origin: static
Description: None
Protocol: Port based, Layer 3 interface: None
IP addresses: None
STP: None, RTG: None.
Number of interfaces: Tagged 0 (Active = 0), Untagged 0 (Active = 0)
None

```

show vlans (Q-in-Q Tunneling)

```

user@switch> show vlans dot1q-tunneling
Name      Tag      Interfaces
sv100     100      xe-0/0/4.0*, xe-0/0/15.0*

```

show vlans extensive (Q-in-Q Tunneling)

```

user@switch> show vlans sv100 extensive
VLAN: sv100, Created at: Sat Sep 10 12:53:52 2011
802.1Q Tag: 100, Internal index: 2, Admin State: Enabled, Origin: Static
Dot1q Tunneling Status: Enabled
Customer VLAN ranges:
    10-20
    40-50
Protocol: Port Mode
Number of interfaces: Tagged 1 (Active = 1), Untagged 0 (Active = 0)
    ge-0/0/0.0, tagged, trunk

Number of mapping rules:
    Push 1 (Active = 0), Policy 0 (Active = 0), Swap 0 (Active = 0)

    xe-0/0/3.0*, 300, push

```

show vlans extensive (Q-in-Q Tunneling and L2TP)

```

user@switch> show vlans v1 extensive
VLAN: v1, Created at: Fri Mar 2 05:07:38 2012
802.1Q Tag: 100, Internal index: 4, Admin State: Enabled, Origin: Static
Dot1q Tunneling status: Enabled
Layer2 Protocol Tunneling status: Enabled

```

show vlans sort-by tag

```

user@switch> show vlans sort-by tag
Name      Tag      Interfaces
default   None
__vlan-x_1__  1

```


__vlan-x_2__	2	None
__vlan-x_3__	3	None
__vlan-x_4__	4	None
__vlan-x_5__	5	None
__vlan-x_6__	6	None
__vlan-x_7__	7	None
__vlan-x_8__	8	None
__vlan-x_9__	9	None
__vlan-x_10__	10	None
__vlan-x_11__	11	None
__vlan-x_12__	12	None
__vlan-x_13__	13	None
__vlan-x_14__	14	None
__vlan-x_15__	15	None
__vlan-x_16__	16	None
__vlan-x_17__	17	None
__vlan-x_18__	18	None
__vlan-x_19__	19	None
__vlan-x_20__	20	None

show vlans sort-by name

```
user@switch> show vlans sort-by employee
```

Name	Tag	Interfaces
__employee_120__	120	xe-0/0/22.0*
__employee_121__	121	xe-0/0/22.0*
__employee_122__	122	xe-0/0/22.0*
__employee_123__	123	xe-0/0/22.0*
__employee_124__	124	xe-0/0/22.0*
__employee_125__	125	xe-0/0/22.0*
__employee_126__	126	xe-0/0/22.0*
__employee_127__	127	xe-0/0/22.0*

```
__employee_128__ 128    xe-0/0/22.0*
__employee_129__ 129    xe-0/0/22.0*
__employee_130__ 130    xe-0/0/22.0*
__employee_130__ 130    xe-0/0/22.0*
```

show vlans tag

```
user@switch> show vlans employee
```

Name	Tag	Interfaces
__employee_120__	120	xe-0/0/22.0*
__employee_121__	121	xe-0/0/22.0*
__employee_122__	122	xe-0/0/22.0*
__employee_123__	123	xe-0/0/22.0*
__employee_124__	124	xe-0/0/22.0*
__employee_125__	125	xe-0/0/22.0*
__employee_126__	126	xe-0/0/22.0*
__employee_127__	127	xe-0/0/22.0*
__employee_128__	128	xe-0/0/22.0*
__employee_129__	129	xe-0/0/22.0*
__employee_130__	130	xe-0/0/22.0*

CHAPTER 19

MAC Address Operational Commands

- `show ethernet-switching mac-learning-log`
- `show ethernet-switching mac-notification`
- `show ethernet-switching statistics aging`
- `show ethernet-switching statistics mac-learning`

show ethernet-switching mac-learning-log

Syntax	show ethernet-switching mac-learning-log
Release Information	Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	Displays the event log of learned MAC addresses.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • show ethernet-switching table on page 387 • show ethernet-switching interfaces on page 383
List of Sample Output	show ethernet-switching mac-learning-log on page 406
Output Fields	Table 38 on page 406 lists the output fields for the show ethernet-switching mac-learning-log command. Output fields are listed in the approximate order in which they appear.

Table 38: show ethernet-switching mac-learning-log Output Fields

Field Name	Field Description
Date and Time	Timestamp in UTC when the MAC operation occurred.
vlan_name	VLAN name. A value defined by the user for all user-configured VLANs. The name of the VLAN on which the MAC is learned.
MAC	Learned MAC address.
Event op	MAC address that are added, learned, deleted, changed or moved from one interface to another interface.
Interface Name	The name of the interface on which the MAC address is learned. When a MAC address is moved, there is another field with the name of the interface. The log displays the name of the interface from where the MAC address moved, and the name of the interface to where the MAC address moved.
Flags	Displays the MAC address flags in which the MAC event occurred. This option is for debugging purposes.

Sample Output

show ethernet-switching mac-learning-log

```

user@switch> show ethernet-switching mac-learning-log
Mon Jun 30 13:49:49 2014 vlan_name v11+11 mac 00:10:94:00:00:02 was learned on
ge-1/0/22.0 with flags: 0x2001f << MAC address that as dynamically learned
Mon Jun 30 13:50:29 2014 vlan_name v11+11 mac 00:10:94:00:00:02 was deleted from
ge-1/0/22.0 with flags: 0x1080 << MAC address that was deleted
Mon Jun 30 13:51:28 2014 vlan_name v11+11 mac 00:00:00:01:01:01 was added to
ge-1/0/22.0 with flags: 0x2013f << Static MAC address that was added
Mon Jun 30 13:51:46 2014 vlan_name v11+11 mac 00:00:00:01:01:01 was deleted from

```

```
ge-1/0/22.0 with flags: 0x1120 << delete of Static MAC address that was deleted
Mon Jun 30 13:52:03 2014 vlan_name v11+11 mac 00:10:94:00:00:02 was learned on
ge-1/0/22.0 with flags: 0x2001f << MAC address that was dynamically learned
Mon Jun 30 13:52:11 2014 vlan_name v11+11 mac 00:10:94:00:00:02 was moved from
ge-1/0/22.0 to ge-1/0/21.0 with flags: 0x2101f << MAC address that was moved
Mon Jun 30 13:54:24 2014 vlan_name v11+11 mac 00:10:94:00:00:02 was changed on
ge-1/0/21.0 with flags: 0x2113f << MAC address that changed from a dynamic
address to a static address
```

show ethernet-switching mac-notification

Syntax	show ethernet-switching mac-notification
Release Information	Command introduced in Junos OS Release 9.6 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	Display information about MAC notification.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • <i>Verifying That MAC Notification Is Working Properly</i>
List of Sample Output	show ethernet-switching mac-notification (MAC Notification Enabled) on page 408 show ethernet-switching mac-notification (MAC Notification Disabled) on page 408
Output Fields	Table 39 on page 408 lists the output fields for the show ethernet-switching mac-notification command. Output fields are listed in the order in which they appear.

Table 39: show ethernet-switching mac-notification Output Fields

Field Name	Field Description
Notification Status	MAC notification status: <ul style="list-style-type: none"> • Enabled—MAC notification is enabled. • Disabled—MAC notification is disabled.
Notification Interval	MAC notification interval in seconds.
Notifications Sent	Number of notifications sent to SNMP when MACs are learned or when MACs age out.
Notifications Table Maxsize	Maximum size of the notification table, which is populated when notifications are sent to the SNMP server.

Sample Output

show ethernet-switching mac-notification (MAC Notification Enabled)

```

user@switch> show ethernet-switching mac-notification
Notification Status           : Enabled
Notification Interval         : 30
Notifications Sent            : 0
Notifications Table Maxsize   : 256

```

Sample Output

show ethernet-switching mac-notification (MAC Notification Disabled)

```

user@switch> show ethernet-switching mac-notification
Notification Status           : Disabled
Notification Interval         : 0

```

Notifications Sent : 0
Notifications Table Maxsize : 256

show ethernet-switching statistics aging

Syntax	show ethernet-switching statistics aging <brief detail>
Release Information	Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	Display media access control (MAC) aging statistics.
Options	none —(Optional) Display MAC aging statistics. brief detail —(Optional) Display the specified level of output.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none"> • show ethernet-switching statistics mac-learning on page 412 • mac-table-aging-time on page 318 • <i>Configuring MAC Table Aging</i>
List of Sample Output	show ethernet-switching statistics aging on page 411
Output Fields	Table 40 on page 410 lists the output fields for the show ethernet-switching statistics aging command. Output fields are listed in the approximate order in which they appear.

Table 40: show ethernet-switching statistics aging Output Fields

Field Name	Field Description	Level of Output
Total age messages received	Total number of aging messages received from the hardware.	All levels
Immediate aging	Aging message indicating that the entry should be removed immediately.	All levels
MAC address seen	Aging message indicating that the MAC address has been detected by hardware and that the aging timer should be stopped.	All levels
MAC address not seen	Aging message indicating that the MAC address has not been detected by the hardware and that the aging timer should be started.	All levels
Error age messages	The received aging message contains the following errors: <ul style="list-style-type: none"> • Invalid VLAN—The VLAN of the packet does not exist. • No such entry—The MAC address and VLAN pair provided by the aging message does not exist. • Static entry—An unsuccessful attempt was made to age out a static MAC entry. 	All levels

Sample Output

show ethernet-switching statistics aging

```
user@switch> show ethernet-switching statistics aging
```

```
Total age messages received: 0
```

```
Immediate aging: 0, MAC address seen: 0, MAC address not seen: 0
```

```
Error age messages: 0
```

```
Invalid VLAN: 0, No such entry: 0, Static entry: 0
```

show ethernet-switching statistics mac-learning

Syntax	<code>show ethernet-switching statistics mac-learning</code> <code><brief detail></code> <code><interface <i>interface-name</i>></code>
Release Information	Command introduced in Junos OS Release 9.4 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	Display media access control (MAC) learning statistics.
Options	none —(Optional) Display MAC learning statistics for all interfaces. brief detail —(Optional) Display the specified level of output. The default is brief . interface <i>interface-name</i> —(Optional) Display MAC learning statistics for the specified interface.
Required Privilege Level	view
Related Documentation	<ul style="list-style-type: none">• show ethernet-switching statistics aging• show ethernet-switching mac-learning-log• show ethernet-switching table• show ethernet-switching interfaces• Example: Setting Up Basic Bridging and a VLAN for an EX Series Switch• Example: Setting Up Bridging with Multiple VLANs for EX Series Switches• show ethernet-switching statistics aging on page 410• show ethernet-switching mac-learning-log on page 406• show ethernet-switching table on page 387• show ethernet-switching interfaces on page 383• Example: Setting Up Basic Bridging and a VLAN on the QFX Series on page 43• Example: Setting Up Bridging with Multiple VLANs
List of Sample Output	show ethernet-switching statistics mac-learning on page 413 show ethernet-switching statistics mac-learning detail on page 414 show ethernet-switching statistics mac-learning interface ge-0/0/28 detail on page 414 show ethernet-switching statistics mac-learning interface on page 414 show ethernet-switching statistics mac-learning detail (QFX Series) on page 414
Output Fields	Table 41 on page 413 lists the output fields for the show ethernet-switching statistics mac-learning command. Output fields are listed in the approximate order in which they appear.

Table 41: show ethernet-switching statistics mac-learning Output Fields

Field Name	Field Description	Level of Output
Interface	Name of the interface for which statistics are being reported. (Displayed in the output under the heading Interface .)	All levels
Learning message from local packets	MAC learning message generated due to packets coming in on the management interface. (Displayed in the output under the heading Local pkts .)	All levels
Learning message from transit packets	MAC learning message generated due to packets coming in on network interfaces. (Displayed in the output under the heading Transit pkts .)	All levels
Learning message with error	<p>MAC learning messages received with errors (Displayed under the heading Error):</p> <ul style="list-style-type: none"> • Invalid VLAN—The VLAN of the packet does not exist. • Invalid MAC—The MAC address is either NULL or a multicast MAC address. • Security violation—The MAC address is not an allowed MAC address. • Interface down—The MAC address is learned on an interface that is down. • Incorrect membership—The MAC address is learned on an interface that is not a member of the VLAN. • Interface limit—The number of MAC addresses learned on the interface has exceeded the limit. • MAC move limit—This MAC address has moved among multiple interfaces too many times in a given interval. • VLAN limit—The number of MAC addresses learned on the VLAN has exceeded the limit. • VLAN membership limit—The number of MAC addresses learned on the interface as a member of the specified VLAN (VLAN membership MAC limit) has exceeded the limit. • Invalid VLAN index—The VLAN of the packet, although configured, does not yet exist in the kernel. • Interface not learning—The MAC address is learned on an interface that does not yet allow learning—for example, the interface is blocked. • No nexthop—The MAC address is learned on an interface that does not have a unicast next hop. • MAC learning disabled—The MAC address is learned on an interface on which MAC learning has been disabled. • Others—The message contains some other error. 	All levels

Sample Output

show ethernet-switching statistics mac-learning

```

user@switch> show ethernet-switching statistics mac-learning

Learning stats: 0 learn msg rcvd, 0 error
  Interface      Local pkts      Transit pkts      Error
  ge-0/0/0.0      0                0                0
  ge-0/0/1.0      0                0                0
  ge-0/0/2.0      0                0                0
  ge-0/0/3.0      0                0                0

```

show ethernet-switching statistics mac-learning detail

```

user@switch> show ethernet-switching statistics mac-learning detail
Learning stats: 0 learn msg rcvd, 0 error

Interface: ge-0/0/0.0
Learning message from local packets: 0
Learning message from transit packets: 1
Learning message with error: 0
  Invalid VLAN: 0      Invalid MAC: 0
  Security violation: 0    Interface down: 0
  Incorrect membership: 0  Interface limit: 0
  MAC move limit: 0      VLAN limit: 0
  Invalid VLAN index: 0   Interface not learning: 0
  No nexthop: 0          MAC learning disabled: 0
  Others: 0

```

```

Interface: ge-0/0/1.0
Learning message from local packets: 0
Learning message from transit packets: 2
Learning message with error: 0
  Invalid VLAN: 0      Invalid MAC: 0
  Security violation: 0    Interface down: 0
  Incorrect membership: 0  Interface limit: 0
  MAC move limit: 0      VLAN limit: 0
  Invalid VLAN index: 0   Interface not learning: 0
  No nexthop: 0          MAC learning disabled: 0
  Others: 0

```

show ethernet-switching statistics mac-learning interface ge-0/0/28 detail

```

user@switch> show ethernet-switching statistics mac-learning interface ge-0/0/28 detail

Interface: ge-0/0/28.0
Learning message from local packets: 0
Learning message from transit packets: 5
Learning message with error: 0
  Invalid VLAN: 0      Invalid MAC: 0
  Security violation: 0    Interface down: 0
  Incorrect membership: 0  Interface limit: 0
  MAC move limit: 0      VLAN limit: 0
                          VLAN membership limit: 20
  Invalid VLAN index: 0   Interface not learning: 0
  No nexthop: 0          MAC learning disabled: 0
  Others: 0

```

show ethernet-switching statistics mac-learning interface

```

user@switch> show ethernet-switching statistics mac-learning interface ge-0/0/1

```

Interface	Local pkts	Transit pkts	Error
ge-0/0/1.0	0	1	1

show ethernet-switching statistics mac-learning detail (QFX Series)

```

user@switch> show ethernet-switching statistics mac-learning detail
Learning stats: 0 learn msg rcvd, 0 error

Interface: xe-0/0/0.0
Learning message from local packets: 0
Learning message from transit packets: 1
Learning message with error: 0

```

Invalid VLAN:	0	Invalid MAC:	0
Security violation:	0	Interface down:	0
Incorrect membership:	0	Interface limit:	0
MAC move limit:	0	VLAN limit:	0
Invalid VLAN index:	0	Interface not learning:	0
No nexthop:	0	MAC learning disabled:	0
Others:	0		

Interface: xe-0/0/1.0

Learning message from local packets: 0

Learning message from transit packets: 2

Learning message with error: 0


Invalid VLAN:	0	Invalid MAC:	0
Security violation:	0	Interface down:	0
Incorrect membership:	0	Interface limit:	0
MAC move limit:	0	VLAN limit:	0
Invalid VLAN index:	0	Interface not learning:	0
No nexthop:	0	MAC learning disabled:	0
Others:	0		

CHAPTER 20

Spanning Tree Monitoring Commands

- clear spanning-tree statistics
- show spanning-tree bridge
- show spanning-tree interface
- show spanning-tree mstp configuration
- show spanning-tree statistics

clear spanning-tree statistics

List of Syntax	Syntax on page 418 Syntax (EX Series Switches and the QFX Series) on page 418
Syntax	clear spanning-tree statistics <interface <i>interface-name</i> > <logical-system <i>logical-system-name</i> >
Syntax (EX Series Switches and the QFX Series)	clear spanning-tree statistics <interface <i>interface-name</i> >
Release Information	Command introduced in Junos OS Release 8.4. Command introduced in Junos OS Release 9.0 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	Clear Spanning Tree Protocol statistics.
Options	none —Reset STP counters for all interfaces for all routing instances. interface <i>interface-name</i> —(Optional) Clear STP statistics for the specified interface only. logical-system <i>logical-system-name</i> —(Optional) Clear STP statistics on a particular logical system.
<div> NOTE: The logical-system option is not available on QFabric systems.</div>	
Required Privilege Level	clear
Related Documentation	<ul style="list-style-type: none">show spanning-tree statistics on page 432
List of Sample Output	clear stp statistics on page 418

Sample Output

clear stp statistics

```
user@host> clear stp statistics
```


show spanning-tree bridge

List of Syntax [Syntax on page 419](#)
[Syntax \(QFX Series\) on page 419](#)

Syntax show spanning-tree bridge
 <brief | detail>
 <msti *msti-id*>
 <routing-instance *routing-instance-name*>
 <vlan-id *vlan-id*>

Syntax (QFX Series) show spanning-tree bridge
 <brief | detail>
 <msti *msti-id*>
 <vlan-id *vlan-id*>

Release Information Command introduced in Junos OS Release 8.4.
 Command introduced in Junos OS Release 11.1 for the QFX Series.

Description Display the configured or calculated Spanning Tree Protocol (STP) parameters.

Options **none**—(Optional) Display brief STP bridge information for all multiple spanning-tree instances (MSTIs).

brief | detail—(Optional) Display the specified level of output.

msti *msti-id*—(Optional) Display STP bridge information for the specified MSTI.

routing-instance *routing-instance-name*—(Optional) Display STP bridge information for the specified routing instance.

vlan-id *vlan-id*—(Optional) Display STP bridge information for the specified VLAN.

Required Privilege Level view

List of Sample Output [show spanning-tree bridge routing-instance on page 420](#)
[show spanning-tree bridge msti on page 421](#)
[show spanning-tree bridge vlan-id \(MSTP\) on page 422](#)
[show spanning-tree bridge \(RSTP\) on page 422](#)
[show spanning-tree bridge vlan-id \(RSTP\) on page 423](#)

Output Fields [Table 42 on page 419](#) lists the output fields for the **show spanning-tree bridge** command. Output fields are listed in the approximate order in which they appear.

Table 42: show spanning-tree bridge Output Fields

Field Name	Field Description
Routing instance name	Name of the routing instance under which the bridge is configured.
Enabled protocol	Spanning Tree Protocol type enabled.

Table 42: show spanning-tree bridge Output Fields (*continued*)

Field Name	Field Description
Root ID	Bridge ID of the elected spanning-tree root bridge. The bridge ID consists of a configurable bridge priority and the MAC address of the bridge.
Root cost	Calculated cost to reach the root bridge from the bridge where the command is entered.
Root port	Interface that is the current elected root port for this bridge.
CIST regional root	Bridge ID of the elected MSTP regional root bridge.
CIST internal root cost	Calculated cost to reach the regional root bridge from the bridge where the command is entered.
Hello time	Configured number of seconds between transmissions of configuration bridge protocol data units (BPDUs).
Maximum age	Configured maximum expected arrival time of hello bridge protocol data units (BPDUs).
Forward delay	How long an STP bridge port remains in the listening and learning states before transitioning to the forwarding state.
Hop count	Configured maximum number of hops a BPDU can be forwarded in the MSTP region.
Message age	Number of elapsed seconds since the most recent BPDU was received.
Number of topology changes	Total number of STP topology changes detected since the routing device last booted.
Time since last topology change	Number of elapsed seconds since the most recent topology change.
Bridge ID (Local)	Locally configured bridge ID. The bridge ID consists of a configurable bridge priority and the MAC address of the bridge.
Extended system ID	System identifier.
MSTI regional root	Bridge ID of the elected MSTP regional root bridge.

Sample Output

show spanning-tree bridge routing-instance

```

user@host> show spanning-tree bridge routing-instance vs1 detail
STP bridge parameters
Routing instance name       : vs1
Enabled protocol           : MSTP

```

```

STP bridge parameters for CIST
  Root ID                : 32768.00:13:c3:9e:c8:80
  Root cost               : 0
  Root port              : ge-10/2/0
  CIST regional root     : 32768.00:13:c3:9e:c8:80
  CIST internal root cost : 22000
  Hello time             : 2 seconds
  Maximum age            : 20 seconds
  Forward delay          : 15 seconds
  Hop count              : 18
  Message age            : 0
  Number of topology changes : 1
  Time since last topology change : 1191 seconds
  Local parameters
    Bridge ID             : 32768.00:90:69:0b:7f:d1
    Extended system ID    : 1

STP bridge parameters for MSTI 1
  MSTI regional root     : 32769.00:13:c3:9e:c8:80
  Root cost              : 22000
  Root port              : ge-10/2/0
  Hello time             : 2 seconds
  Maximum age            : 20 seconds
  Forward delay          : 15 seconds
  Hop count              : 18
  Number of topology changes : 1
  Time since last topology change : 1191 seconds
  Local parameters
    Bridge ID             : 32769.00:90:69:0b:7f:d1
    Extended system ID    : 1

STP bridge parameters for MSTI 2
  MSTI regional root     : 32770.00:13:c3:9e:c8:80
  Root cost              : 22000
  Root port              : ge-10/2/0
  Hello time             : 2 seconds
  Maximum age            : 20 seconds
  Forward delay          : 15 seconds
  Hop count              : 18
  Number of topology changes : 1
  Time since last topology change : 1191 seconds
  Local parameters
    Bridge ID             : 32770.00:90:69:0b:7f:d1
    Extended system ID    : 1

```

show spanning-tree bridge msti

```

user@host> show spanning-tree bridge msti 1 routing-instance vs1 detail
STP bridge parameters
Routing instance name      : vs1
Enabled protocol          : MSTP

STP bridge parameters for MSTI 1
  MSTI regional root     : 32769.00:13:c3:9e:c8:80
  Root cost              : 22000
  Root port              : xe-10/2/0
  Hello time             : 2 seconds
  Maximum age            : 20 seconds
  Forward delay          : 15 seconds
  Hop count              : 18

```

```
Number of topology changes      : 1
Time since last topology change : 1191 seconds
Local parameters
  Bridge ID                     : 32769.00:90:69:0b:7f:d1
  Extended system ID            : 1
```

show spanning-tree bridge vlan-id (MSTP)

```
user@host> show spanning-tree bridge vlan-id 1101 routing-instance vs1 detail
```

```
STP bridge parameters
Routing instance name          : vs1
Enabled protocol               : MSTP

STP bridge parameters for CIST
Root ID                       : 32768.00:13:c3:9e:c8:80
Root cost                     : 0
Root port                     : xe-10/2/0
CIST regional root            : 32768.00:13:c3:9e:c8:80
CIST internal root cost       : 22000
Hello time                    : 2 seconds
Maximum age                   : 20 seconds
Forward delay                 : 15 seconds
Hop count                     : 18
Message age                   : 0
Number of topology changes     : 0
Local parameters
  Bridge ID                   : 32768.00:90:69:0b:7f:d1
  Extended system ID          : 1
  Hello time                  : 2 seconds
  Maximum age                 : 20 seconds
  Forward delay               : 15 seconds
  Path cost method            : 32 bit
  Maximum hop count           : 20
```

show spanning-tree bridge (RSTP)

```
user@host> show spanning-tree bridge
```

```
STP bridge parameters
Routing instance name          : GLOBAL
Enabled protocol               : RSTP
Root ID                       : 28672.00:90:69:0b:3f:d0
Hello time                    : 2 seconds
Maximum age                   : 20 seconds
Forward delay                 : 15 seconds
Message age                   : 0
Number of topology changes     : 58
Time since last topology change : 14127 seconds
Local parameters
  Bridge ID                   : 28672.00:90:69:0b:3f:d0
  Extended system ID          : 0

STP bridge parameters for bridge VLAN 10
Root ID                       : 28672.00:90:69:0b:3f:d0
Hello time                    : 2 seconds
Maximum age                   : 20 seconds
Forward delay                 : 15 seconds
Message age                   : 0
Number of topology changes     : 58
Time since last topology change : 14127 seconds
Local parameters
  Bridge ID                   : 28672.00:90:69:0b:3f:d0
```

```

Extended system ID          : 0

STP bridge parameters for bridge VLAN 20
Root ID                     : 28672.00:90:69:0b:3f:d0
Hello time                   : 2 seconds
Maximum age                  : 20 seconds
Forward delay                : 15 seconds
Message age                  : 0
Number of topology changes   : 58
Time since last topology change : 14127 seconds
Local parameters
  Bridge ID                  : 28672.00:90:69:0b:3f:d0
  Extended system ID         : 0

```

show spanning-tree bridge vlan-id (RSTP)

```

user@host> show spanning-tree bridge vlan-id 10
STP bridge parameters
Routing instance name        : GLOBAL
Enabled protocol             : RSTP

STP bridge parameters for VLAN 10
Root ID                     : 28672.00:90:69:0b:3f:d0
Hello time                   : 2 seconds
Maximum age                  : 20 seconds
Forward delay                : 15 seconds
Message age                  : 0
Number of topology changes   : 58
Time since last topology change : 14127 seconds
Local parameters
  Bridge ID                  : 28672.00:90:69:0b:3f:d0
  Extended system ID         : 0

```

show spanning-tree interface

List of Syntax	Syntax on page 424 Syntax (EX Series Switches and the QFX Series) on page 424
Syntax	<pre>show spanning-tree interface <brief detail> <msti <i>msti-id</i>> <routing-instance <i>routing-instance-name</i>> <vlan-id <i>vlan-id</i>></pre>
Syntax (EX Series Switches and the QFX Series)	<pre>show spanning-tree interface <brief detail> <msti <i>msti-id</i>> <vlan-id <i>vlan-id</i>></pre>
Release Information	<p>Command introduced in Junos OS Release 8.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p>
Description	Display the configured or calculated interface-level STP parameters.
Options	<p>none—Display brief STP interface information.</p> <p>brief detail—(Optional) Display the specified level of output.</p> <p>msti <i>msti-id</i>—(Optional) Display STP interface information for the specified MST instance.</p> <p>routing-instance <i>routing-instance-name</i>—(Optional) Display STP interface information for the specified routing instance.</p> <p>vlan-id <i>vlan-id</i>—(Optional) Display STP interface information for the specified VLAN.</p>
Required Privilege Level	view
List of Sample Output	show spanning-tree interface on page 425 show spanning-tree interface (QFX Series) on page 426 show spanning-tree interface detail on page 426 show spanning-tree interface msti on page 428 show spanning-tree interface vlan-id on page 428 show spanning-tree interface (VSTP) on page 429 show spanning-tree interface vlan-id (VSTP) on page 429
Output Fields	<p>Table 43 on page 424 lists the output fields for the show spanning-tree interface command. Output fields are listed in the approximate order in which they appear.</p>

Table 43: show spanning-tree Interface Output Fields

Field Name	Field Description
Interface name	Interface configured to participate in the STP, RSTP, VSTP, or MSTP instance.

Table 43: show spanning-tree Interface Output Fields (*continued*)

Field Name	Field Description
Port ID	Logical interface identifier configured to participate in the MSTP or VSTP instance.
Designated port ID	Port ID of the designated port for the LAN segment to which this interface is attached.
Designated bridge ID	Bridge ID of the designated bridge for the LAN segment to which this interface is attached.
Port Cost	Configured cost for the interface.
Port State	STP port state: forwarding (FWD), blocking (BLK), listening, learning, or disabled.
Port Role	MSTP, VSTP, or RSTP port role: designated (DESG), backup (BKUP), alternate (ALT), (ROOT), or Root Prevented (Root-Prev).
Link type	MSTP, VSTP, or RSTP link type. Shared or point-to-point (pt-pt) and edge or nonedge.
Alternate	Identifies the interface as an MSTP, VSTP, or RSTP alternate root port (Yes) or nonalternate root port (No).
Boundary Port	Identifies the interface as an MSTP regional boundary port (Yes) or nonboundary port (No).

Sample Output

show spanning-tree interface

```
user@host> show spanning-tree interface routing-instance vs1 detail
Spanning tree interface parameters for instance 0
```

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ae1	128:1	128:1	32768.0090690b47d1	1000	FWD	DESG
ge-2/1/2	128:2	128:2	32768.0090690b47d1	20000	FWD	DESG
ge-2/1/5	128:3	128:3	32768.0090690b47d1	29999	FWD	DESG
ge-2/2/1	128:4	128:26	32768.0013c39ec880	20000	FWD	ROOT
xe-9/2/0	128:5	128:5	32768.0090690b47d1	2000	FWD	DESG
xe-9/3/0	128:6	128:6	32768.0090690b47d1	2000	FWD	DESG

```
Spanning tree interface parameters for instance 1
```

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ae1	128:1	128:1	32769.0090690b47d1	1000	FWD	DESG
ge-2/1/2	128:2	128:2	32769.0090690b47d1	20000	FWD	DESG
ge-2/1/5	128:3	128:3	32769.0090690b47d1	29999	FWD	DESG
ge-2/2/1	128:4	128:26	32769.0013c39ec880	20000	FWD	ROOT
xe-9/2/0	128:5	128:5	32769.0090690b47d1	2000	FWD	DESG
xe-9/3/0	128:6	128:6	32769.0090690b47d1	2000	FWD	DESG

Spanning tree interface parameters for instance 2

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ae1	128:1	128:1	32770.0090690b47d1	1000	FWD	DESG
ge-2/1/2	128:2	128:2	32770.0090690b47d1	20000	FWD	DESG
ge-2/1/5	128:3	128:3	32770.0090690b47d1	29999	FWD	DESG
ge-2/2/1	128:4	128:26	32770.0013c39ec880	20000	FWD	ROOT
xe-9/2/0	128:5	128:5	32770.0090690b47d1	2000	FWD	DESG
xe-9/3/0	128:6	128:6	32770.0090690b47d1	2000	FWD	DESG

show spanning-tree interface (QFX Series)

```
user@1f0> show spanning-tree interface routing-instance vs1 detail
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ae1	128:1	128:1	32768.0090690b47d1	1000	FWD	DESG
ge-2/1/2	128:2	128:2	32768.0090690b47d1	20000	FWD	DESG
ge-2/1/5	128:3	128:3	32768.0090690b47d1	29999	FWD	DESG
ge-2/2/1	128:4	128:26	32768.0013c39ec880	20000	FWD	ROOT
xe-9/2/0	128:5	128:5	32768.0090690b47d1	2000	FWD	DESG
xe-9/3/0	128:6	128:6	32768.0090690b47d1	2000	FWD	DESG

Spanning tree interface parameters for instance 1

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ae1	128:1	128:1	32769.0090690b47d1	1000	FWD	DESG
ge-2/1/2	128:2	128:2	32769.0090690b47d1	20000	FWD	DESG
ge-2/1/5	128:3	128:3	32769.0090690b47d1	29999	FWD	DESG
ge-2/2/1	128:4	128:26	32769.0013c39ec880	20000	FWD	ROOT
xe-9/2/0	128:5	128:5	32769.0090690b47d1	2000	FWD	DESG
xe-9/3/0	128:6	128:6	32769.0090690b47d1	2000	FWD	DESG

Spanning tree interface parameters for instance 2

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ae1	128:1	128:1	32770.0090690b47d1	1000	FWD	DESG
ge-2/1/2	128:2	128:2	32770.0090690b47d1	20000	FWD	DESG
ge-2/1/5	128:3	128:3	32770.0090690b47d1	29999	FWD	DESG
ge-2/2/1	128:4	128:26	32770.0013c39ec880	20000	FWD	ROOT
xe-9/2/0	128:5	128:5	32770.0090690b47d1	2000	FWD	DESG
xe-9/3/0	128:6	128:6	32770.0090690b47d1	2000	FWD	DESG

show spanning-tree interface detail

```
user@host> show spanning-tree interface routing-instance vs1 detail
```

Spanning tree interface parameters for instance 0

```
Interface name           : ae1
Port identifier          : 128.1
Designated port ID      : 128.1
Port cost                 : 1000
Port state               : Forwarding
Designated bridge ID     : 32768.00:90:69:0b:47:d1
Port role                 : Designated
Link type                 : Pt-Pt/NONEDGE
```



```

Boundary port                : No

Interface name                : ge-2/1/2
Port identifier               : 128.2
Designated port ID           : 128.2
Port cost                     : 20000
Port state                    : Forwarding
Designated bridge ID         : 32768.00:90:69:0b:47:d1
Port role                     : Designated
Link type                     : Pt-Pt/NONEDGE
Boundary port                : No

Interface name                : ge-2/1/5
Port identifier               : 128.3
Designated port ID           : 128.3
Port cost                     : 29999
Port state                    : Forwarding
Designated bridge ID         : 32768.00:90:69:0b:47:d1
Port role                     : Designated
Link type                     : Pt-Pt/NONEDGE
Boundary port                : No

Interface name                : ge-2/2/1
Port identifier               : 128.4
Designated port ID           : 128.26
Port cost                     : 20000
Port state                    : Forwarding
Designated bridge ID         : 32768.00:13:c3:9e:c8:80
Port role                     : Root
Link type                     : Pt-Pt/NONEDGE
Boundary port                : No

Interface name                : xe-9/2/0
Port identifier               : 128.5
Designated port ID           : 128.5
Port cost                     : 2000
Port state                    : Forwarding
Designated bridge ID         : 32768.00:90:69:0b:47:d1
Port role                     : Designated
Link type                     : Pt-Pt/NONEDGE
Boundary port                : No

Interface name                : xe-9/3/0
Port identifier               : 128.6
Designated port ID           : 128.6
Port cost                     : 2000
Port state                    : Forwarding
Designated bridge ID         : 32768.00:90:69:0b:47:d1
Port role                     : Designated
Link type                     : Pt-Pt/NONEDGE
Boundary port                : No

```

Spanning tree interface parameters for instance 1

```

Interface name                : ae1
Port identifier               : 128.1
Designated port ID           : 128.1
Port cost                     : 1000
Port state                    : Forwarding
Designated bridge ID         : 32768.00:90:69:0b:47:d1

```

```

Port role           : Designated
Link type           : Pt-Pt/NONEDGE
Boundary port       : No

Interface name      : ge-2/1/2
Port identifier     : 128.2
Designated port ID  : 128.2
Port cost           : 20000
Port state          : Forwarding
Designated bridge ID : 32768.00:90:69:0b:47:d1
Port role           : Designated
Link type           : Pt-Pt/NONEDGE
Boundary port       : No

Interface name      : ge-2/1/5
Port identifier     : 128.3
Designated port ID  : 128.3
Port cost           : 29999
Port state          : Forwarding
Designated bridge ID : 32768.00:90:69:0b:47:d1
Port role           : Designated
Link type           : Pt-Pt/NONEDGE
Boundary port       : No

Interface name      : ge-2/2/1
Port identifier     : 128.4
Designated port ID  : 128.26
Port cost           : 20000
Port state          : Forwarding
Designated bridge ID : 32768.00:13:c3:9e:c8:80
Port role           : Root
Link type           : Pt-Pt/NONEDGE
Boundary port       : No

...

```

show spanning-tree interface msti

```

user@host> show spanning-tree interface msti 1 routing-instance vs1 detail
Spanning tree interface parameters for instance 1

```

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
xe-7/0/0	128:1	128:1	32769.0090690b4fd1	2000	FWD	DESG
ge-5/1/0	128:2	128:2	32769.0090690b4fd1	20000	FWD	DESG
ge-5/1/1	128:3	128:3	32769.0090690b4fd1	20000	FWD	DESG
ae1	128:4	128:1	32769.0090690b47d1	10000	BLK	ALT
ge-5/1/4	128:5	128:3	32769.0090690b47d1	20000	BLK	ALT
xe-7/2/0	128:6	128:6	32769.0090690b47d1	2000	FWD	ROOT

show spanning-tree interface vlan-id

```

user@host> show spanning-tree interface vlan-id 101 routing-instance vs1 detail
Spanning tree interface parameters for instance 0

```

Interface	Port ID	Designated port ID	Designated bridge ID	Port Cost	State	Role
ge-11/0/5	128:1	128:1	32768.0090690b7fd1	20000	FWD	DESG
ge-11/0/6	128:2	128:1	32768.0090690b7fd1	20000	BLK	BKUP
ge-11/1/0	128:3	128:2	32768.0090690b4fd1	20000	BLK	ALT
ge-11/1/1	128:4	128:3	32768.0090690b4fd1	20000	BLK	ALT

ge-11/1/4	128:5	128:1	32768.0090690b47d1	20000	BLK	ALT
xe-10/0/0	128:6	128:5	32768.0090690b4fd1	2000	BLK	ALT
xe-10/2/0	128:7	128:4	32768.0090690b47d1	2000	FWD	ROOT

show spanning-tree interface (VSTP)

```
user@host> show spanning-tree interface
```

Spanning tree interface parameters for instance 0

Interface	Port ID	Designated port ID	Designated bridge ID	Cost	State	Role
ge-1/0/1	128:1	128:1	28672.0090690b3fe0	20000	FWD	DESG
ge-1/0/2	128:2	128:2	28672.0090690b3fe0	20000	FWD	DESG

Spanning tree interface parameters for VLAN 10

Interface	Port ID	Designated port ID	Designated bridge ID	Cost	State	Role
ge-1/0/1	128:1	128:1	28672.0090690b3fe0	20000	FWD	DESG
ge-1/0/2	128:2	128:2	28672.0090690b3fe0	20000	FWD	DESG

Spanning tree interface parameters for VLAN 20

Interface	Port ID	Designated port ID	Designated bridge ID	Cost	State	Role
ge-1/0/1	128:1	128:1	28672.0090690b3fe0	20000	FWD	DESG
ge-1/0/2	128:2	128:2	28672.0090690b3fe0	20000	FWD	DESG

show spanning-tree interface vlan-id (VSTP)

```
user@host> show spanning-tree interface vlan-id 10
```

Spanning tree interface parameters for VLAN 10

Interface	Port ID	Designated port ID	Designated bridge ID	Cost	State	Role
ge-1/0/1	128:1	128:1	28672.0090690b3fe0	20000	FWD	DESG
ge-1/0/2	128:2	128:2	28672.0090690b3fe0	20000	FWD	DESG

show spanning-tree mstp configuration

List of Syntax	Syntax on page 430 Syntax (EX Series Switch and the QFX Series) on page 430
Syntax	show spanning-tree mstp configuration <brief detail> <routing-instance <i>routing-instance-name</i> >
Syntax (EX Series Switch and the QFX Series)	show spanning-tree mstp configuration <brief detail>
Release Information	Command introduced in Junos OS Release 8.4. Command introduced in Junos OS Release 9.0 for EX Series switches. Command introduced in Junos OS Release 11.1 for the QFX Series.
Description	Display the MSTP configuration.
Options	none—Display MSTP configuration information. brief detail—(Optional) Display the specified level of output. routing-instance <i>routing-instance-name</i> —(Optional) Display MSTP configuration information for the specified routing instance.
Required Privilege Level	view
List of Sample Output	show spanning-tree mstp configuration detail on page 431 show spanning-tree mstp configuration detail (QFX Series) on page 431
Output Fields	Table 44 on page 430 lists the output fields for the show spanning-tree mstp configuration command. Output fields are listed in the approximate order in which they appear.

Table 44: show spanning-tree mstp configuration Output Fields

Field Name	Field Description
Context id	Internally generated identifier.
Region name	MSTP region name carried in the MSTP BPDUs.
Revision	Revision number of the MSTP configuration.
Configuration digest	Numerical value derived from the VLAN-to-instance mapping table.
MSTI	MST instance identifier.
Member VLANs	VLAN identifiers associated with the MSTI.

Sample Output

show spanning-tree mstp configuration detail

```
user@host> show spanning-tree mstp configuration routing-instance vs1 detail
MSTP configuration information
Context identifier      : 1
Region name            : henry
Revision               : 3
Configuration digest    : 0x6da4b5c4fd587757eef35675365e1

MSTI      Member VLANs
0 0-99,101-199,201-4094
1 100
2 200
```

show spanning-tree mstp configuration detail (QFX Series)

```
user@1f0> show spanning-tree mstp configuration routing-instance vs1 detail
MSTP configuration information
Context identifier      : 1
Region name            : henry
Revision               : 3
Configuration digest    : 0x6da4b5c4fd587757eef35675365e1

MSTI      Member VLANs
0 0-99,101-199,201-4094
1 100
2 200
```

show spanning-tree statistics

List of Syntax	Syntax on page 432 Syntax (EX Series Switch and the QFX Series) on page 432
Syntax	<pre>show spanning-tree statistics <brief detail> <interface <i>interface-name</i>> <routing-instance <i>routing-instance-name</i>></pre>
Syntax (EX Series Switch and the QFX Series)	<pre>show spanning-tree statistics <brief detail> <interface <i>interface-name</i> vlan <i>vlan-id</i>></pre>
Release Information	<p>Command introduced in Junos OS Release 8.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>Command introduced in Junos OS Release 11.1 for QFX Series switches.</p>
Description	Display STP statistics.
Options	<p>none—Display brief STP statistics.</p> <p>brief detail—(Optional) Display the specified level of output.</p> <p>interface <i>interface-name</i>—(Optional) Display STP statistics for the specified interface.</p> <p>routing-instance <i>routing-instance-name</i>—(Optional) Display STP statistics for the specified routing instance.</p>
Required Privilege Level	view
List of Sample Output	show spanning-tree statistics routing-instance on page 433 show spanning-tree statistics interface routing-instance detail on page 433
Output Fields	<p>Table 45 on page 432 lists the output fields for the show spanning-tree statistics command. Output fields are listed in the approximate order in which they appear.</p>

Table 45: show spanning-tree statistics Output Fields

Field Name	Field Description
Message type	Type of message being counted.
BPDUs sent	Total number of BPDUs sent.
BPDUs received	Total number of BPDUs received.
BPDUs sent in last interval	Number of BPDUs sent within a specified interval.
BPDUs received in last interval	Number of BPDUs received within a specified interval.

Table 45: show spanning-tree statistics Output Fields (*continued*)

Field Name	Field Description
Interface	Interface for which the statistics are being displayed.
Next BPDU transmission	Number of seconds until the next BPDU is scheduled to be sent.

Sample Output

show spanning-tree statistics routing-instance

```

user@host> show spanning-tree statistics routing-instance vs1 detail
Routing instance level STP statistics
Message type           : bpdus
BPDUs sent             : 1396
BPDUs received         : 1027
BPDUs sent in last interval : 5      (duration: 4 sec)
BPDUs received in last interval: 4    (duration: 4 sec)

```

show spanning-tree statistics interface routing-instance detail

```

user@host> show spanning-tree statistics interface ge-11/1/4 routing-instance vs1 detail
Interface  BPDUs sent  BPDUs received  Next BPDU
                                     transmission
ge-11/1/4      7           190           0

```

